Text as Data

Meeting 4: Dictionaries and Validation

Petro Tolochko

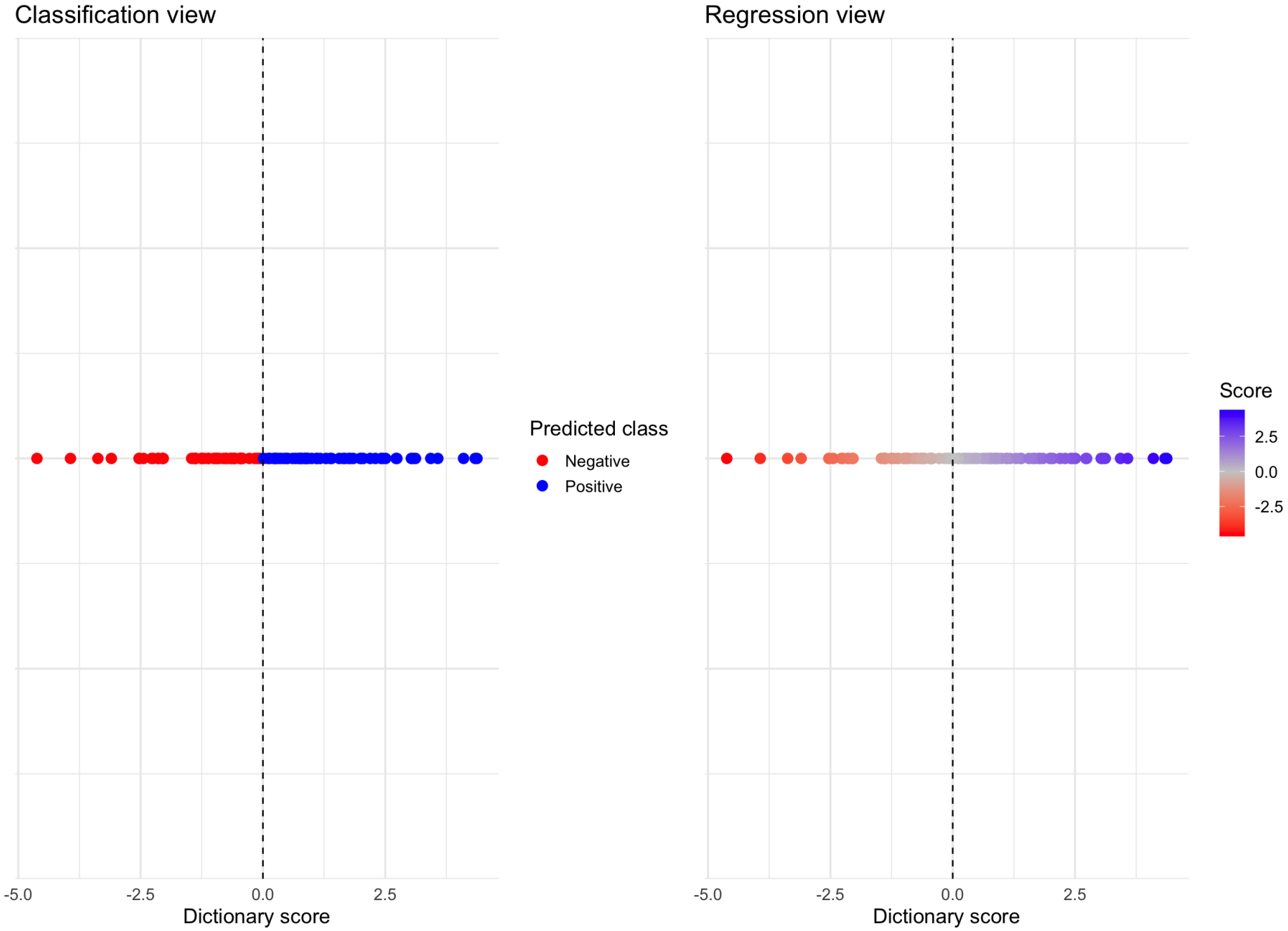
Dictionaries

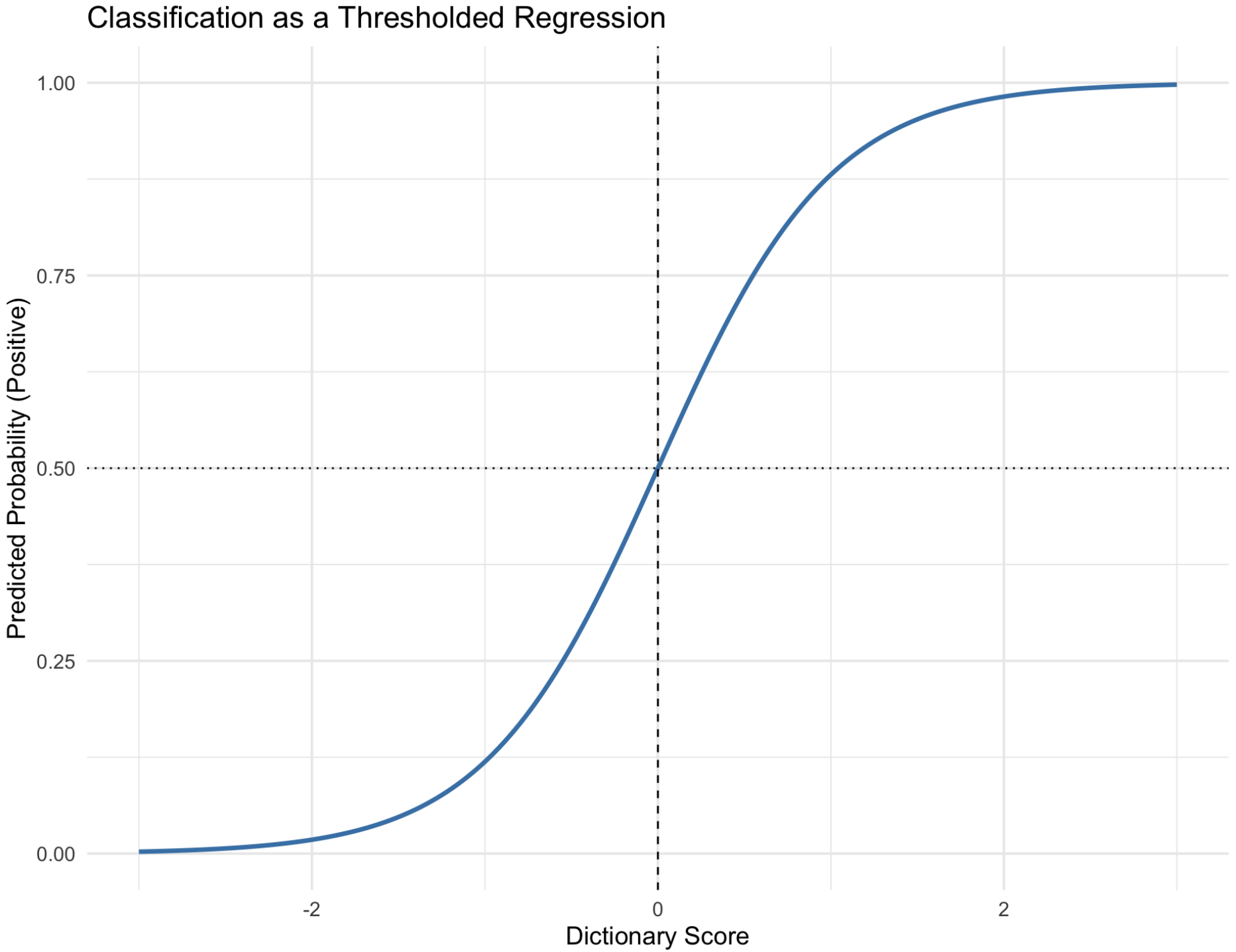
- Rule-based method
- List of words (or phrases) that indicate a category
- Create your own or use/edit existing dictionaries

Use cases for dictionaries

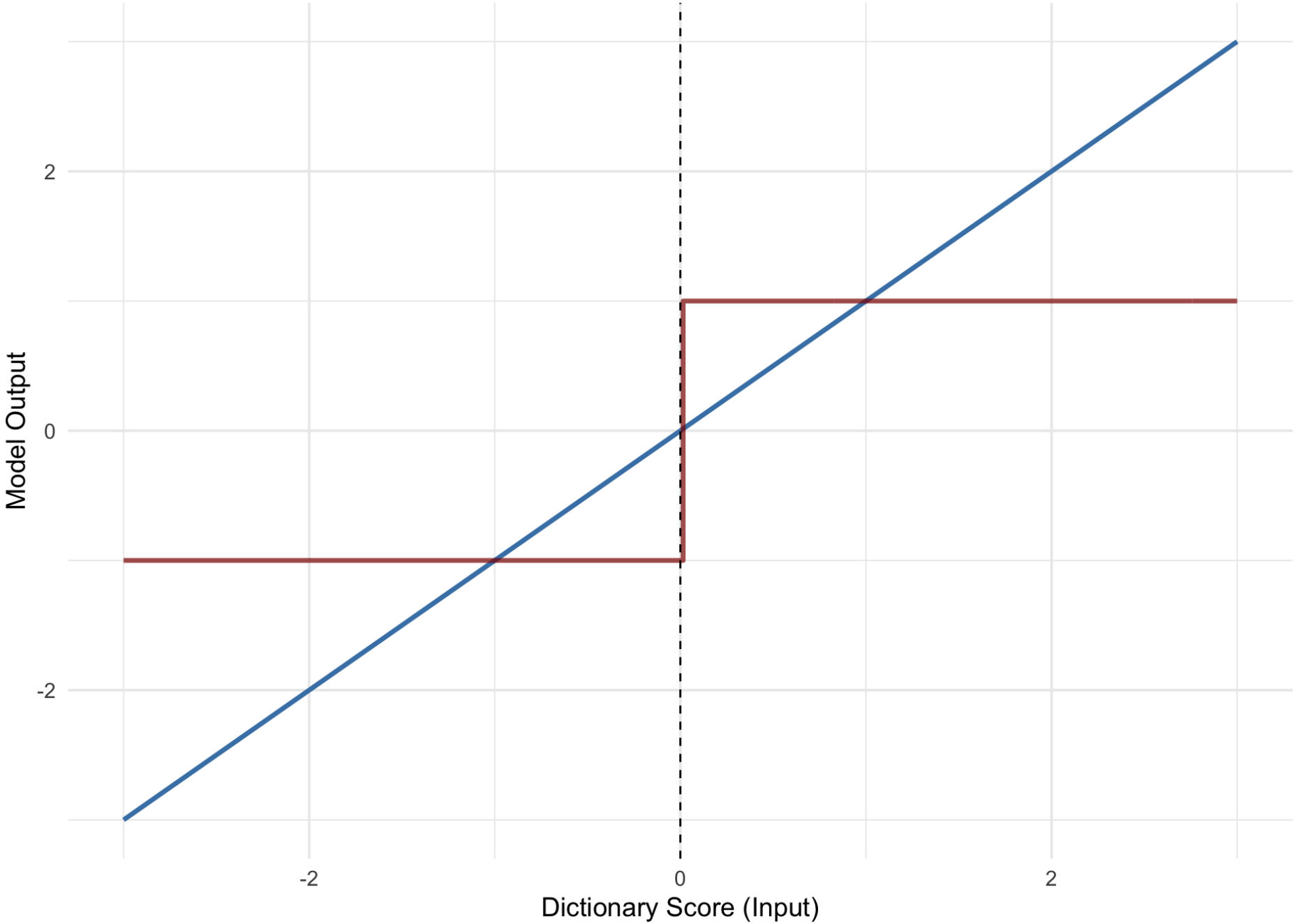
- Classification
- Regression
- Search string (form of classification)

Classification vs. Regression


Classification view
Regression view
Predicted class
Negative
Positive
Score
2.5
0.0
-2.5
-5.0
-2.5
0.0
2.5
Dictionary score
-5.0
-2.5
0.0
2.5
Dictionary score


Classification as a Thresholded Regression
1.00
0.75
0.50
0.25
0.00
Predicted Probability (Positive)
-2
0
2
Dictionary Score

Classification vs. Regression



Blue = continuous regression output | Red = discrete classification

Classification Examples

- Classify texts into:
  - positive/negative
  - populist/non-populist
  - related/unrelated to a certain category

Dictionary for Classification

Let D^+ be a set of positive words, D^- a set of negative words. Given a text $t = (w_1, \ldots, w_n)$:

$$\text{pos}(t) = \sum_{i=1}^{n} \mathbf{1}[w_i \in D^+], \qquad \text{neg}(t) = \sum_{i=1}^{n} \mathbf{1}[w_i \in D^-]$$

$$\hat{y}(t) = \begin{cases} \text{positive} & \text{if pos}(t) > \text{neg}(t) \\ \text{negative} & \text{if neg}(t) > \text{pos}(t) \\ \text{neutral} & \text{otherwise} \end{cases}$$

Regression Examples

- "Score" texts on some dimension:
  - Positive/negative
  - Emotional content (anger, sadness, etc.)

Dictionary for Regression

Each word w has an associated weight $s(w)$, e.g., $s(\text{“excellent”}) = +3$, $s(\text{“terrible”}) = -3$.

Score of a text $t = (w_1, \ldots, w_n)$:

$$\text{score}(t) = \sum_{i=1}^{n} s(w_i)$$

Properties:

- Positive words push score upward, negative words downward.
- Magnitude reflects **intensity of sentiment**.
- Can normalize by length:

$$\hat{s}(t) = \frac{\text{score}(t)}{n}$$

Classification + Regression Combined

Dictionary score as a continuous variable, thresholded for classification:

$$\text{score}(t) = \sum_{i=1}^{n} s(w_i)$$

$$\hat{y}(t) = \begin{cases} \text{positive} & \text{if score}(t) > 0 \\ \text{negative} & \text{if score}(t) < 0 \\ \text{neutral} & \text{if score}(t) = 0 \end{cases}$$

Key insight: Classification is just thresholding a regression score.

Dictionary Scoring

Given a text $t = (w_1, w_2, \ldots, w_n)$ and dictionary D:

$$s_D(t) = \sum_{i=1}^{n} \mathbf{1}[w_i \in D]$$

- $s_D(t) =$ raw dictionary count
- Binary classification:

$$y(t) = \begin{cases} 1 & \text{if } s_D(t) > \tau \\ 0 & \text{otherwise} \end{cases}$$

- Continuous regression score:

$$y(t) = \frac{s_D(t)}{n}$$

(normalized by text length)

Normalization Approaches

Raw counts can be misleading (longer texts $\Rightarrow$ higher scores).

- Per-token normalization:

$$\hat{s}_D(t) = \frac{s_D(t)}{|t|}$$

- Per-1000 words:

$$\hat{s}_D(t) = \frac{s_D(t)}{|t|} \times 1000$$

- TF-IDF weighting (dictionary as term set):

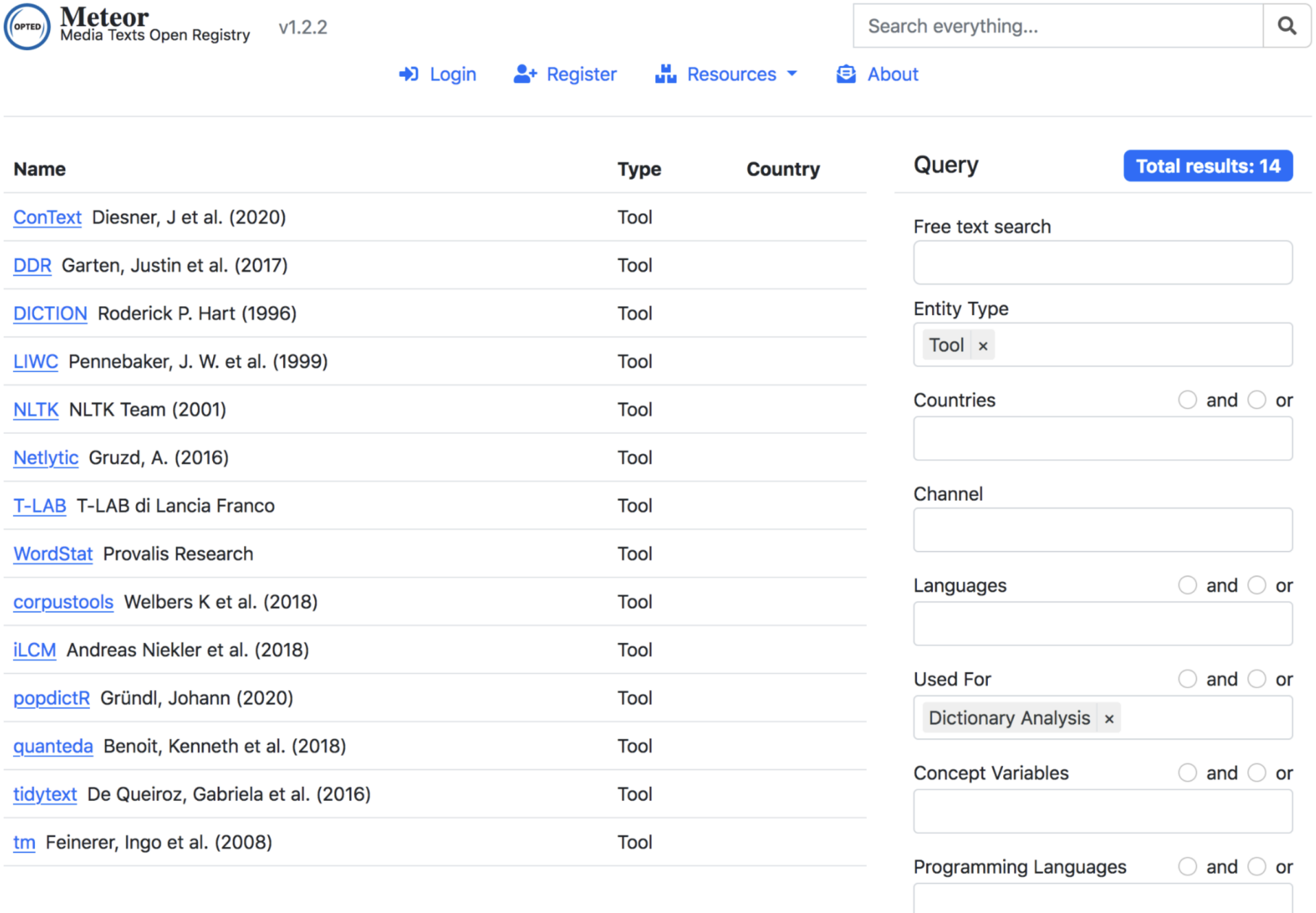
$$\text{tf-idf}(w,t) = \text{tf}(w,t) \cdot \log \frac{N}{df(w)}$$

Existing vs. Own Dictionaries

Existing vs. Own Dictionaries

- Many existing and validated dictionaries:

METEOR


Meteor
Media Texts Open Registry
v1.2.2
Search everything...
Login
Register
Resources
About
Name
Type
Country
ConText Diesner, J et al. (2020) Tool
DDR Garten, Justin et al. (2017) Tool
DICTION Roderick P. Hart (1996) Tool
LIWC Pennebaker, J. W. et al. (1999) Tool
NLTK NLTK Team (2001) Tool
Netlytic Gruzd, A. (2016) Tool
T-LAB T-LAB di Lancia Franco Tool
WordStat Provalis Research Tool
corpustools Welbers K et al. (2018) Tool
iLCM Andreas Niekler et al. (2018) Tool
popdictR Gründl, Johann (2020) Tool
quanteda Benoit, Kenneth et al. (2018) Tool
tidytext De Queiroz, Gabriela et al. (2016) Tool
tm Feinerer, Ingo et al. (2008) Tool
Query
Total results: 14
Free text search
Entity Type
Tool ×
Countries and or
Channel
Languages and or
Used For and or
Dictionary Analysis ×
Concept Variables and or
Programming Languages and or

Existing vs. Own Dictionaries

- Many existing and validated dictionaries:
- Many instances of creating ad-hoc dictionaries:

Heidenreich et al., 2020; Lind et al., 2020

Table 3 Boolean search strings used for retrieval of migration-related news articles

Country	Language	Search string
Spain	Spanish	asilo* OR inmigra* OR refugiad* OR migrante* OR migratori* OR "sin papeles" OR "campo de desplazados" OR patera* OR emigra* OR "libre circulación" OR "fuga de cerebros"
UK	English	asyl* OR immigrant* OR immigrat* OR migrant* OR migrat* OR refugee* OR foreigner* OR "undocumented worker*" OR "guest worker*" OR "foreign worker*" OR emigrat* OR "freedom of movement" OR "free movement"
Germany	German	asyl* OR immigrant* OR immigriert* OR immigrat* OR migrant* OR migrat* OR flüchtling* OR ausländer* OR zuwander* OR zugewander* OR einwander* OR eingewander* OR gastarbeiter* OR "ausländische arbeitnehmer*" OR emigr* OR auswander* OR ausgewander* OR personenfreizügigkeit* OR arbeitnehmerfreizügigkeit* OR "freier personenverkehr*"

Dictionary pros vs cons

Pros

- Often needed to select data (search strings)
- High reliability and control
- High transparency and reproducibility

Cons

- Difficulty increases with the latency of the construct
- Language nuances

Questions?

Validation in Dictionary Analysis

- Source and data selection determines results and conclusions
- Are the selected data sources and selected data points representative for your target concept or discourse?
- Relevant?
- Representative?

Relevance of search string validation

- Sampling based on search strings popular (Stryker et al., 2016) and recommended (Barberá et al., 2021)
- Reviews of search string validation procedures
  - out of 83 content analyses, 39% stated the search terms they used, and only 6% discussed their validity (Stryker et al., 2016)
  - out of 105 content analysis studies, 73.3% stated the search terms they used, only 12.4% reported validity metrics (Mahl et al., 2022)
- Careless application of non-validated search terms may lead to noisy inferences (Mahl et al., 2022)

Key validation approach

- How close is an automated measurement to a more trusted measurement:
  - Human understanding of text

Dictionary validation with manually created baseline

- Code a subset manually (consider intercoder reliability)
- Compare manual decisions with automated classification decisions (via recall, precision, F1)
- Iterative dictionary improvement
- Ideally: manual coding and dictionary development is performed by different persons

Creation of a manual baseline

- Codebook creation
- Who codes manually?
  - Expert coders: Coder recruitment and training sessions
  - Crowdcoders: test questions, majority choice
- Quality assessment: e.g., Inter-coder reliability of involved coders, majority vote
- How reliable? Consider valid disagreement (Baden et al., 2023)
- Documents selected for baseline should be representative for target discourse (e.g., random selection or artificial week)

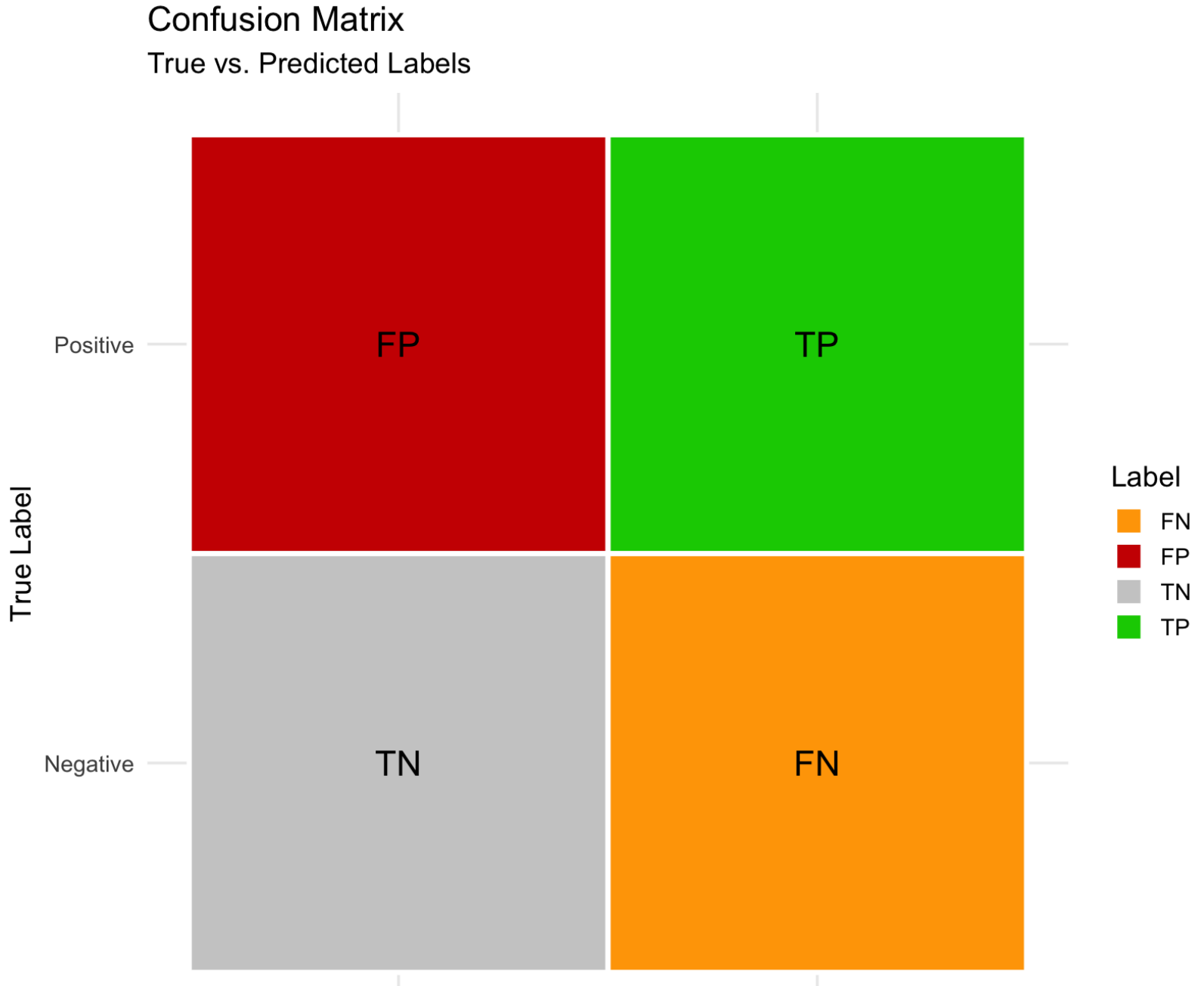
Recall, precision, F1

- Metrics frequently used to express the validity of a search string & more generally also of automated classification methods
- Precision (P)
- Recall (R)
- $F_1 = 2 \cdot \frac{P \cdot R}{P + R}$

$$\text{Precision} = \frac{TP}{TP + FP} \qquad \text{Recall} = \frac{TP}{TP + FN}$$

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \qquad F_\beta = (1 + \beta^2) \cdot \frac{\text{Precision} \cdot \text{Recall}}{\beta^2 \cdot \text{Precision} + \text{Recall}}$$

Confusion Matrix

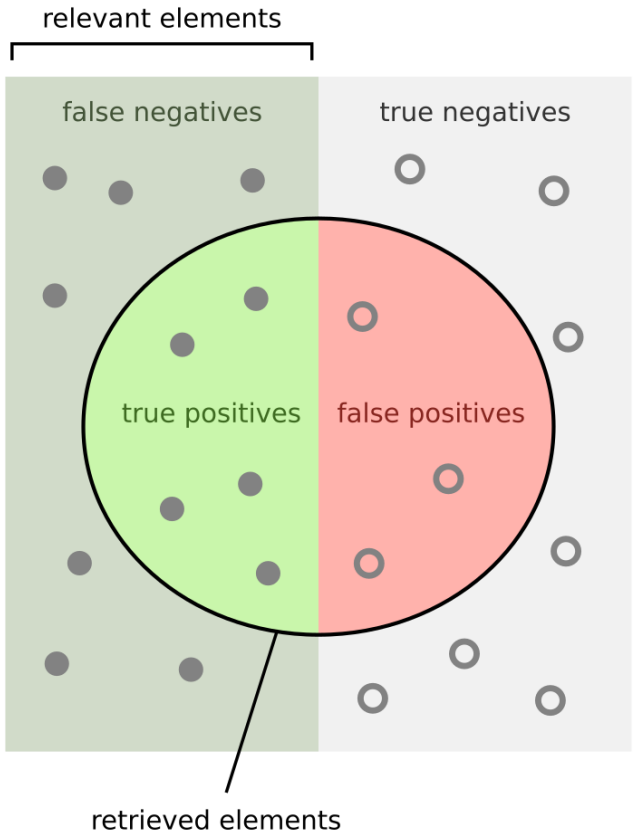


Validation: Confusion Matrix

	Predicted Positive	Predicted Negative
True Positive	TP	FN
True Negative	FP	TN

- Precision: $P = \frac{TP}{TP+FP}$
- Recall: $R = \frac{TP}{TP+FN}$
- F1-score: $F_1 = \frac{2PR}{P+R}$

Precision & Recall



How many retrieved items are relevant?

$$\text{Precision} = \frac{\text{true positives}}{\text{retrieved elements}}$$

How many relevant items are retrieved?

$$\text{Recall} = \frac{\text{true positives}}{\text{relevant elements}}$$

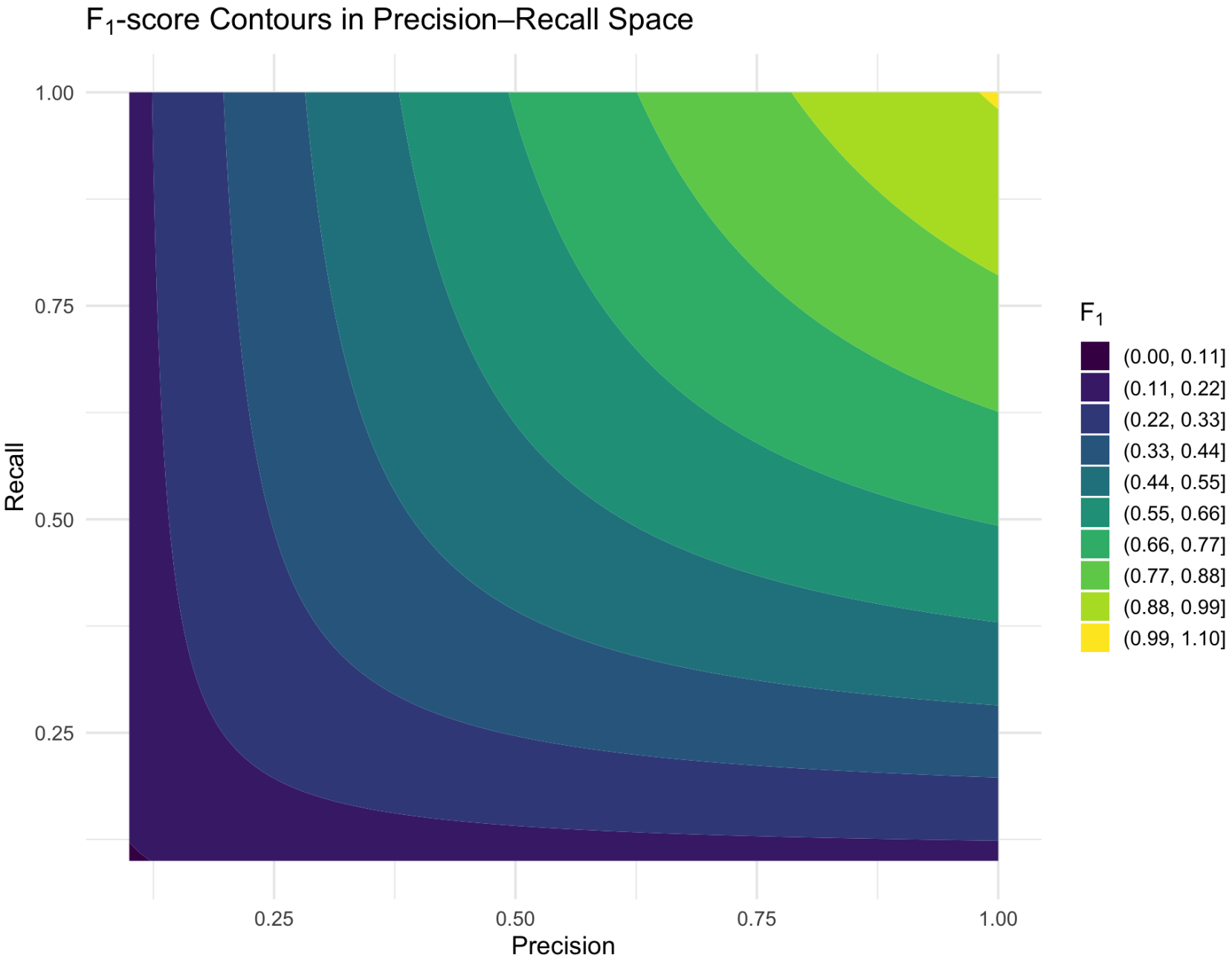
Understanding the F_1-Score

Definition:

$$F_1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}}$$

- Harmonic mean of Precision and Recall.
- Balances *correctness* (Precision) and *coverage* (Recall).
- $F_1 = 1$ only if both Precision and Recall are 1.
- Penalizes large imbalances between the two metrics.

F1 Contour


F_1-score Contours in Precision–Recall Space
Recall
Precision
1.00
0.75
0.50
0.25
0.25
0.50
0.75
1.00
F_1
(0.00, 0.11]
(0.11, 0.22]
(0.22, 0.33]
(0.33, 0.44]
(0.44, 0.55]
(0.55, 0.66]
(0.66, 0.77]
(0.77, 0.88]
(0.88, 0.99]
(0.99, 1.10]

Interpretation in Precision–Recall Space:

- Each pair (P, R) defines a point in the Precision–Recall plane.
- Lines of constant F_1 (iso-F_1 contours) represent equal trade-offs:

$$R = \frac{F_1 \cdot P}{2P - F_1}, \quad P > \frac{F_1}{2}$$

- Higher F_1 values lie near the upper-right corner (both P and R high).
- The optimal decision threshold is where your PR curve touches the highest contour.

ROC and Precision–Recall (PR) Curves

Goal: Visualize trade-offs between correct and incorrect classifications as the decision threshold changes.

1. ROC Curve (Receiver Operating Characteristic)

$$\text{True Positive Rate (TPR)} = \frac{TP}{TP + FN}, \quad \text{False Positive Rate (FPR)} = \frac{FP}{FP + TN}$$

- Plots TPR (y-axis) vs. FPR (x-axis)
- Area under the curve (AUC) measures overall discriminative ability
- Random classifier: diagonal line (AUC $= 0.5$)

2. Precision–Recall (PR) Curve

$$\text{Precision} = \frac{TP}{TP + FP}, \quad \text{Recall (TPR)} = \frac{TP}{TP + FN}$$

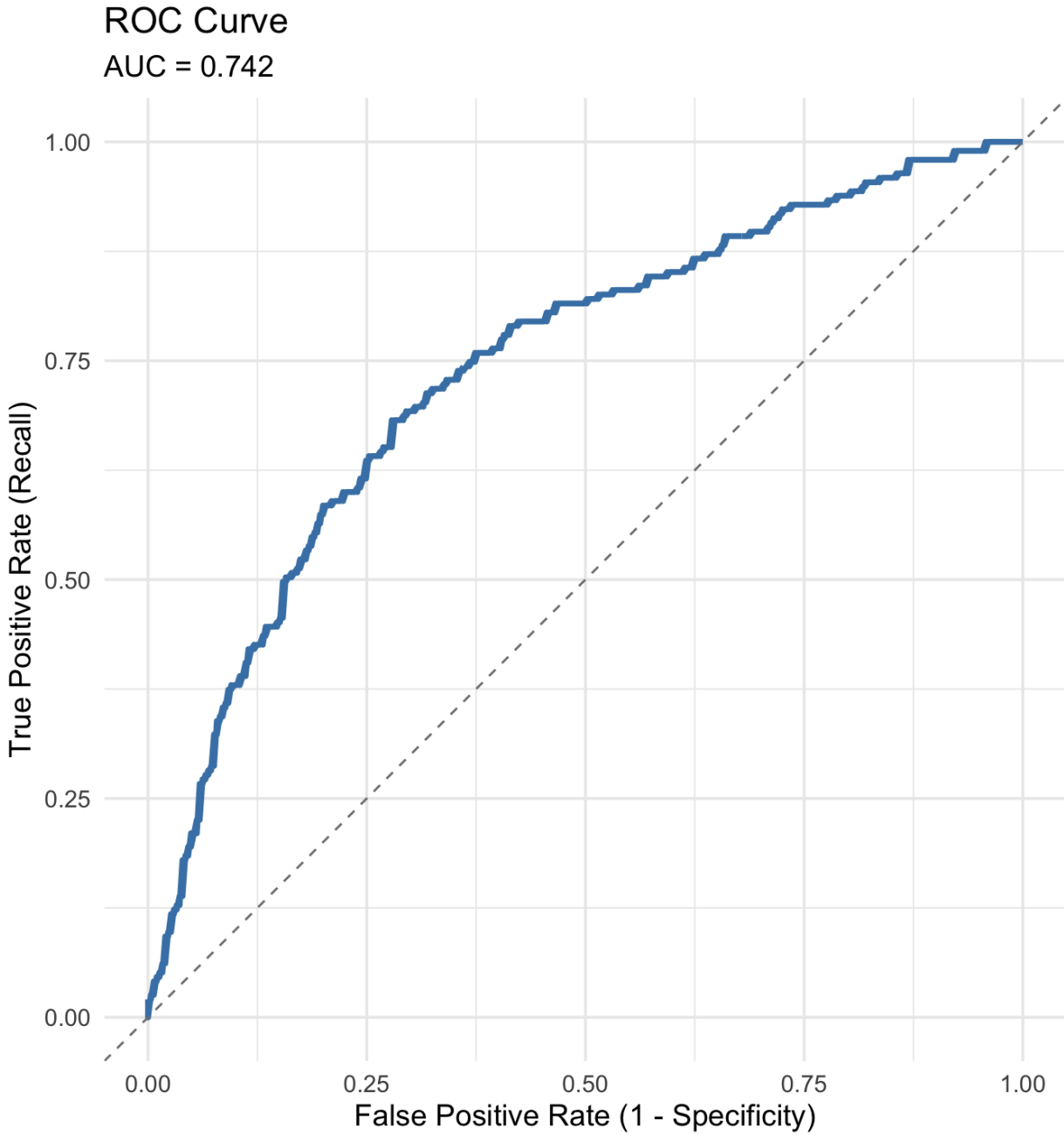
- Plots Precision (y-axis) vs. Recall (x-axis)
- More informative when classes are imbalanced
- F_1-score is highest near the top-right region

ROC & PR Curve

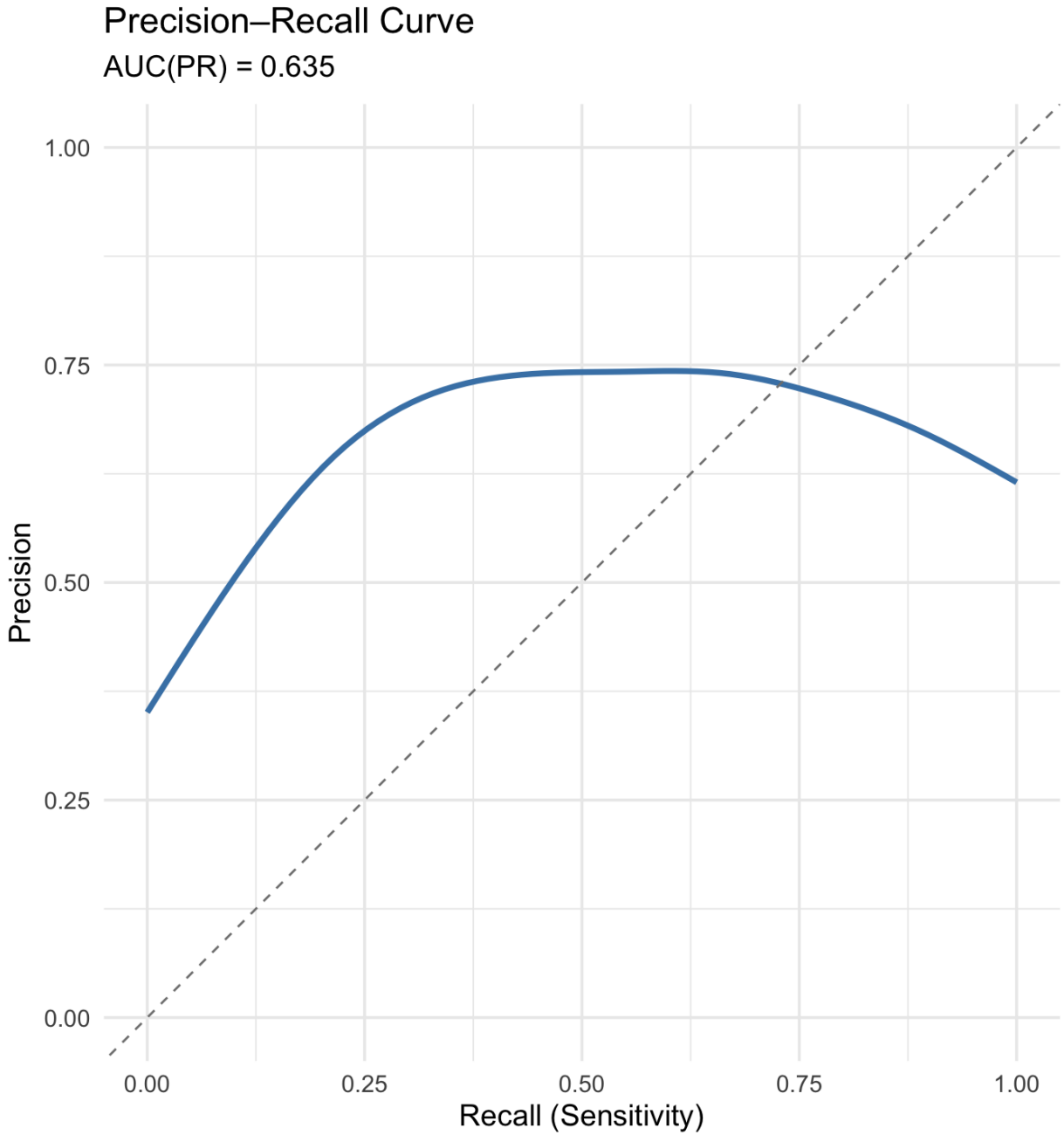
Interpretation:

- ROC: How well positives are ranked above negatives
- PR: How well predicted positives are truly positive

ROC Curve Tradeoff



PR Tradeoff


Precision–Recall Curve
AUC(PR) = 0.635
1.00
0.75
0.50
0.25
0.00
Precision
0.00
0.25
0.50
0.75
1.00
Recall (Sensitivity)

Regular Expressions (regex)

- How to pronounce?
- gɪf
- dʒɪf

Regular Expressions (regex)

- How to pronounce?
- gɪf
- dʒɪf
- rɛ.gɛks
- rɛ.dʒɛks

Regular Expressions (regex)

- How to pronounce?
- **gɪf**
- dʒɪf
- **rɛ.gɛks**
- rɛ.dʒɛks

Regular Expressions (regex)

- How to pronounce?
- gɪf
- **dʒɪf** $\Rightarrow$ wrong
- rɛ.gɛks
- **rɛ.dʒɛks** $\Rightarrow$ wrong

regex

- Formal language to specify search strings
- Insanely difficult

regex

- Formal language to specify search strings
- Insanely difficult
- Nobody can remember anything

regex

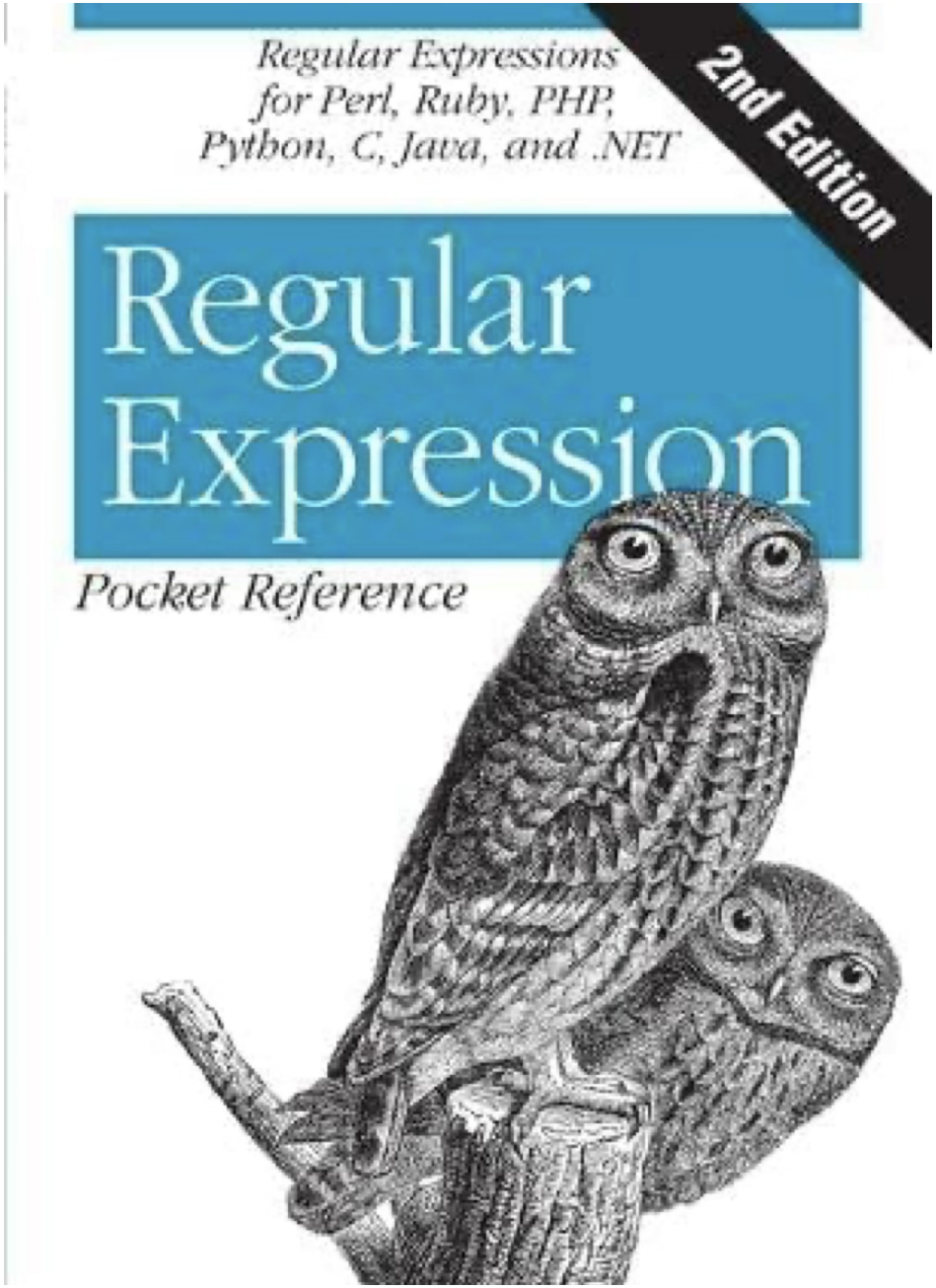
- Formal language to specify search strings
- Insanely difficult
- Nobody can remember anything
- Different flavours

regex

- Formal language to specify search strings
- Insanely difficult
- Nobody can remember anything
- Different flavours

Some people, when confronted with a problem, think “I know, I’ll use regular expressions.” Now they have two problems. — Jamie Zawinski

Regex Book


Regular Expressions
for Perl, Ruby, PHP,
Python, C, Java, and .NET
2nd Edition
Regular
Expression
Pocket Reference

Grimmer / Jurafsky Cheat-sheet

- Disjunctions

RE	Match	Example Patterns Matched
`[mM]oney`	Money or money	"Money"
`[abc]`	'a', 'b', *or* 'c'	"Investing in Iran"
		"is dangerous business"
`[1234567890]`	any digit	"sitting on $7.5 billion dollars"
		"2005 and 2006, more than "
		"$150 million dollars"
`[\.]`	A period	" 'Run!', he screamed."

- Ranges

RE	Match	Example Patterns Matched
`[A-Z]`	an upper case letter	"Rep. Anthony Weiner
		(D-Brooklyn & Queens)"
`[a-z]`	a lower case letter	"ACORN's"
`[0-9]`	a single digit	"(9th CD) "

Grimmer / Jurafsky Cheat-sheet

- Negations

RE	Match	Example Patterns Matched
`[^A-Z]`	not an upper case letter	"ACORN**'s**"
`[^Ss]`	neither 'S' nor 's'	"**ACORN'**s"
`[^\.]`	not a period	" 'Run!', he screamed."

- Optional Characters: `?`, `*`, `+`

RE	Match	Example Patterns Matched
`colou?r`	Words with `u` 0 or 1 times	"color" or "colour "
`oo*h!`	Words with `o` 0 or more times	"oh!" or "ooh!" or "oooh!"
`o+h!`	Words with `o` 1 or more times	"oh!" or "ooh!" or "oooooh!" or

Grimmer / Jurafsky Cheat-sheet

- Start of the line anchor ^, end of the line anchor **$**

RE	Match	Example Patterns Matched
`^[A-Z]`	Upper case start of line	"Palo Alto" "the town of **P**alo Alto"
`^[^A-Z]`	Not upper case start of line	"the town of Palo Alto" "**P**alo Alto"
`^.`	Start of line	"Palo Alto" "the town of Palo Alto"
`.$`	Identify character that ends a line	"Wait**!**" "This is the end**.**"

- "Or"| statements, Useful short hand

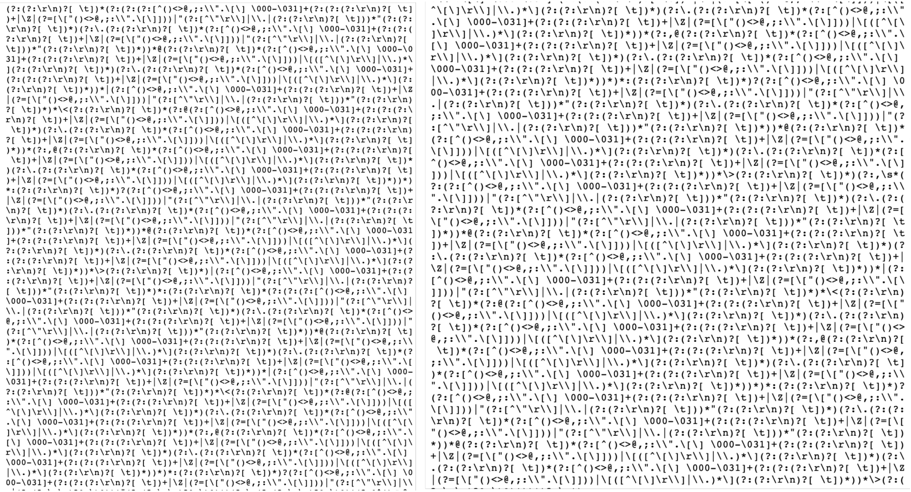
RE	Match	Example Patterns Matched
yours|mine	Matches"yours" or "mine"	"it's either yours or mine"
\ d	Any digit	"1-Mississippi"
\ D	Any non-digit	"1-Mississippi"
\ s	Any whitespace character	"1,_2"
\ S	Any non-whitespace character	"1, 2"
\ w	Any alpha-numeric	"1-Mississippi "
\ W	Any non-alpha numeric	"1-Mississippi"

How difficult to regex an email?

```
(?:[a-z0-9!#$%&'*+/=?^_`{|}~-]+(?:\.[a-z0-9!#$%&'*+/=?^_`{|}~-]+)*|"(?:[\x01-
\x08\x0b\x0c\x0e-\x1f\x21\x23-\x5b\x5d-\x7f]|\\[\x01-\x09\x0b\x0c\x0e-\x7f])*")@(?:
(?:[a-z0-9](?:[a-z0-9-]*[a-z0-9])?\.)+[a-z0-9](?:[a-z0-9-]*[a-z0-9])?|\[(?:(?:(2(5[0-
5]|[0-4][0-9])|1[0-9][0-9]|[1-9]?[0-9]))\.){3}(?:(2(5[0-5]|[0-4][0-9])|1[0-9][0-9]|[1-
9]?[0-9])|[a-z0-9-]*[a-z0-9]:(?:[\x01-\x08\x0b\x0c\x0e-\x1f\x21-\x5a\x53-\x7f]|\\
[\x01-\x09\x0b\x0c\x0e-\x7f])+)\])
```

How difficult to regex an email?

- Rather...



Concepts and Data

- What do we actually want to measure?
- And where do we find it?

Discovery

- Idea from qualitative research
- A “method” to discover new concepts through descriptive analysis
- Grimmer et al., 2022:
  - I. Context Relevance
  - II. No Ground Truth
  - III. Concept vs. Method
  - IV. Data Separation

Principles of Discovery

- **Principle 1: Context relevance.**
- Text as data models complement theory and substantive knowledge. Contextual knowledge amplifies our ability to make computational discoveries

Principles of Discovery

- **Principle 2: No ground truth.**
- There is no ground truth conceptualization; only after a concept is fixed can we talk meaningfully about it being right or wrong

Principles of Discovery

- **Principle 3: Judge the concept, not the method.**
- The method you used to arrive at a conceptualization does not matter for assessing the concept's value – its utility does

Principles of Discovery

- **Principle 4: Separate data is best.**
- Ideally after data is used for discovery it should be discarded in favor of new data for confirming/testing discoveries.

Codebook

- The codebook is the tool you use to code your content
- It is a kind of questionnaire that you use to inquiry the examined texts/photos/videos
- The codebook should be detailed enough so that
  - you can apply it again in the same way after some time (intracoder reliability)
  - other people (with a little training) can also use it in the same way as you (intercoder reliability)

Codebook

- In automated text analysis:
  - Validation
  - Documentation

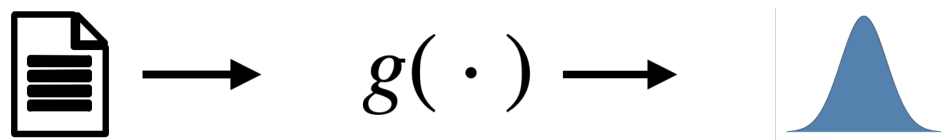
Codebook Development

- Iterative process:
- First draft based on the variables identified in the research question(s)/hypothesis(s).
- Tip: take other studies as a model
- Code material examples using the draft codebook
- Then refine/edit the codebook

Codebook Function

- Used for dimension reduction (e.g., Egami et al., 2022)

Codebook as a function



Probabilistic Interpretation of Metrics

Let $y \in \{0, 1\}$ be the **true class** and $\hat{y} \in \{0, 1\}$ the **predicted class**.

- **Recall (Sensitivity):**

$$R = \frac{TP}{TP + FN} = P(\hat{y} = 1 \mid y = 1)$$

  Probability that a truly positive item is predicted as positive.

- **Precision (Positive Predictive Value):**

$$P = \frac{TP}{TP + FP} = P(y = 1 \mid \hat{y} = 1)$$

  Probability that a predicted positive item is truly positive.

- **F1-score:**

$$F_1 = \frac{2PR}{P + R}$$

  Harmonic mean of precision and recall.

Interpreting the F1 Score

Why the harmonic mean?

- The F1-score balances **precision** (P) and **recall** (R).
- Arithmetic mean:

$$\frac{P+R}{2}$$ (too lenient: a very large value can dominate)

- Harmonic mean:

$$F_1 = \frac{2PR}{P+R}$$

Penalizes imbalance — both P and R must be high.

Properties:

- $F_1 = 1$ only if $P = R = 1$.
- F_1 approaches 0 if either P or R is near 0.
- Generalization: F_β weights recall β times more than precision:

$$F_\beta = (1+\beta^2) \cdot \frac{PR}{\beta^2 P + R}$$

Bayesian Perspective

Let $y = 1$ be the event that the item is truly positive, and $\hat{y} = 1$ the event that the classifier predicts positive.

- **Recall (Sensitivity):**

$$R = P(\hat{y} = 1 \mid y = 1)$$

  Likelihood of predicting positive given a true positive.

- **Precision (Positive Predictive Value):**

$$P = P(y = 1 \mid \hat{y} = 1)$$

  This is a **posterior probability**:

$$P(y = 1 \mid \hat{y} = 1) = \frac{P(\hat{y} = 1 \mid y = 1)\, P(y = 1)}{P(\hat{y} = 1)}$$

- **Interpretation:**
  - Recall $\approx$ quality of the measurement instrument (likelihood).
  - Precision $\approx$ belief update about true label given the prediction (posterior).