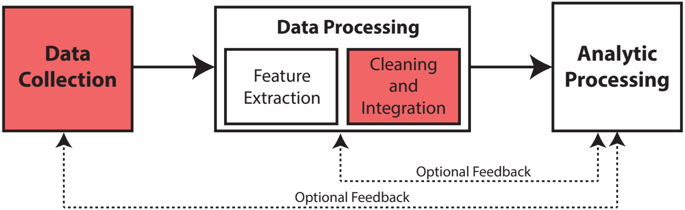
# Data Foundations: Machine Learning Overview and Data Annotation

Instructor: Anthony Rios

# Outline

# Data Science Pipeline




- **Learned** the basics of Python
- **Learned** to process many file types (CSV, JSON, XML)
- **Now** we will discuss **machine learning** and **annotating data**

# Introduction to ML

Discussion on IPad

# Annotated Data

Modern **data science** is driven by **annotated data**.

- In most cases the data we have is the product of **human judgements**.
  - What is the sentiment of the tweet?
  - What is the object in the picture?
  - What is the topic of the news article?

# Issues with human judgement: Ambiguity

- John and Mary are married.

- To each other? or separately?

# Issues with human judgement: Ambiguity

- John and Mary are married.

- To each other? or separately?

# Issues with human judgement: Dogmatism

**Dogmatism** describes the tendency to lay down opinions as **incontrovertibly true**, without respect for conflicting evidence or the opinions of others.

**Which user is more dogmatic in the examples below?**

"I'm supposed to trust the opinion of a MS minion? The people that produced Windows ME, Vista and 8? They don't even understand people, yet they think they can predict the behavior of new, selfguiding AI?"
–anonymous

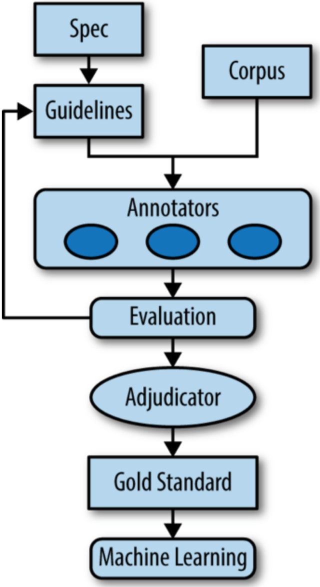
"I think an AI would make it easier for Patients to confide their information because by nature, a robot cannot judge them. Win-win? :D"'
–anonymous

(Fast and Horvitz. 2016)

# Issues with human judgement: Sarcasm

"In many respects, you know, they honor President Obama. He's the founder of ISIS. He's the founder of ISIS. He's the founder. He founded ISIS." – Donald Trump

# Annotation Pipeline




Pustejovsky and Stubbs (2012), Natural Language Annotation for Machine Learning

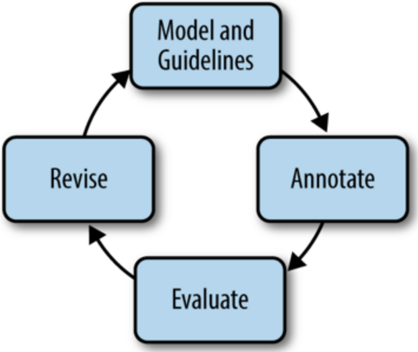
# Annotation Process

1. Determine what to annotate.
2. Formalize the instructions for the annotation task
3. Perform a pilot annotation
4. Annotate the data
5. Compute and report inter-annotator agreement, and release the data.

# Exercise 1

Complete the Sentiment Annotation Survey on Blackboard

# Annotation Pipeline




Pustejovsky and Stubbs (2012), Natural Language Annotation for Machine Learning

# Annotation Guidelines

**Our goal:** Given the constraints of our problem, how can we formalize our descriptions of the annotation process **to encourage multiple annotators to provide the same judgment?**

# Annotation Guidelines

- What is the goal of the project?

- What is each class called and how is it used? (Be specific: provide examples and discuss gray areas)

- What **exactly** should be annotated and what should be left alone?

Pustejovsky and Stubbs (2012), Natural Language Annotation for Machine Learning

# Example: Sentiment

What best describes the speaker's attitude, evaluation, or judgment towards the [target]? If the whole text is a quote from somebody else (original author) and there is no indication of speaker's attitude, then answer below considering the original author as the speaker.

- **Positive**: there is an explicit or implicit clue in the text suggesting that the speaker's attitude or judgment of the [target] is positive (speaker is appreciative, thankful, excited, optimistic, or inspired by the primary entity)
- **Negative**: there is an explicit or implicit clue in the text suggesting that the speaker's attitude or judgment of the [target] is negative (speaker is critical, angry, disappointed in, pessimistic, expressing sarcasm about, or mocking the primary entity)
- **Mixed**: there is an explicit or implicit clue in the text suggesting that the speaker's attitude or judgment of the [target] is both positive and negative.
- **Unknown**: there is no explicit or implicit clue indicating that the speaker feels positively or negatively.

Mohammad 2016

# Practicalities

- Annotation takes time/concentration (can't do it 8 hours a day)

- Annotators get better as they annotate (earlier annotations not as good as later ones)

# Why not do it yourself?

- Expensive/time-consuming
- Multiple people provide a measure of consistency: is the task well enough defined?
- Low agreement = not enough training, guidelines not well enough defined, task is bad.

# Adjudication

- Adjudication is the process of deciding on a single annotation for a piece of text, using information about the **independent annotations**.
- Can be **time-consuming** (or more so) as primary annotation.
- Does **NOT** need to be identical with the primary annotation. (both annotators can be wrong by chance)

# Exercise 2

Judge the annotations in following Data:

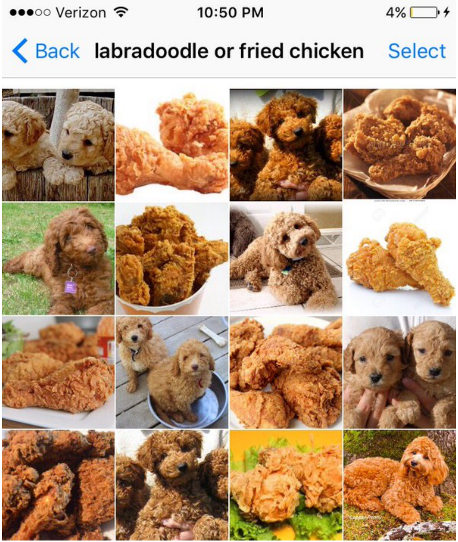
sentiment.txt

You are the judge. Look over the annotations and come up with the following:

- What is your judgment for the correct entity + sentiment annotation?
- How would you amend the annotation guidelines to solicit more consistent annotations?

While judging the annotations, put your judgements in the jupyter notebook file.

# Inter-annotator Agreement




https://twitter.com/teenybiscuit/status/705232709220769792/photo/1

Annotator A (x) vs Annotator B (y)

| | puppy | fried chicken |
|---|---|---|
| puppy | **6** | 3 |
| fried chicken | 2 | **5** |

observed agreement $= 11/16 = 68.75\%$

# Cohen's Kappa

If classes are imbalanced, we can get high inter-annotator agreement simply chance.

Annotator A (x) vs Annotator B (y)

| | puppy | fried chicken |
|---|---|---|
| puppy | **7** | 4 |
| fried chicken | 8 | **81** |

observed agreement $= p_o = 88/100 = 88\%$

# Cohen's Kappa

**Expected probability** ($p_e$) of agreement is how often we would expect two annotators to agree assuming **independent** annotations.

$$p_e = P(A = puppy, B = puppy) + P(A = chicken, B = chicken)$$

$$p_e = P(A = puppy)P(B = puppy) + P(A = chicken)P(B = chicken)$$

# Cohen's Kappa

$$p_e = P(A = puppy)P(B = puppy) + P(A = chicken)P(B = chicken)$$

P(A = puppy) = 15/100 = 0.15
P(B = puppy) = 11/100 = 0.11
P(A = chicken) = 85/100 = 0.85
P(B = chicken) = 89/100 = 0.89

= 0.15*0.11 + 0.85*0.89
= 0.773

Annotator A (x) vs Annotator B (y)

| | puppy | fried chicken |
|---|---|---|
| puppy | **7** | 4 |
| fried chicken | 8 | **81** |

# Cohen's Kappa

If classes are imbalanced, we can get high inter-annotator agreement simply by chance.

$$\kappa = \frac{p_o - p_e}{1 - p_e}$$

$$\kappa = \frac{0.88 - p_e}{1 - p_e}$$

$$\kappa = \frac{0.88 - 0.773}{1 - 0.773}$$

$$= 0.471$$

Annotator A (x) vs Annotator B (y)

| | puppy | fried chicken |
|---|---|---|
| puppy | **7** | 4 |
| fried chicken | 8 | **81** |

$p_o = 88/100 = 0.88\%$

# Cohen's Kappa

"Good" values are subject to interpretation, but rule of thumb

| Score Range | Interpretation |
|---|---|
| 0.80 - 1.00 | Very good agreement |
| 0.60 - 0.80 | Good agreement |
| 0.40 - 0.60 | Moderate agreement |
| 0.20 - 0.40 | Fair agreement |
| $< 0.20$ | Poor agreement |

# Example

Annotator A (x) vs Annotator B (y)

| | puppy | fried chicken |
|---|---|---|
| puppy | **0** | 0 |
| fried chicken | 0 | **100** |

# Exercise 3

Calculate cohen's kappa using the following numbers:

Annotator A (x) vs Annotator B (y)

| | puppy | fried chicken |
|---|---|---|
| puppy | **25** | 25 |
| fried chicken | 25 | **25** |

Annotator A (x) vs Annotator B (y)

| | puppy | fried chicken |
|---|---|---|
| puppy | **50** | 0 |
| fried chicken | 0 | **50** |

| | puppy | fried chicken |
|---|---|---|
| puppy | **0** | 50 |
| fried chicken | 50 | **0** |

# Exercise 4

Write code to calculate and print the cohen's kappa between rater1 and rater2 (the lists below are already in notebook).

rater1 = ['yes','no','yes','yes','yes','yes','no','yes','yes']

rater2 = ['yes','no','no','yes','yes','yes','yes','yes','yes']

Hint: You will need to create the confusion matrix (matrix of how many times each rater agrees for each item). This can be represented as 4 variables

# Issues with Cohen's Kappa

- Cohen's kappa can be used for any number of classes.
- Still requires **two** annotators who evaluate the same items.
- Fleiss' kappa generalizes to **multiple** annotators, each of whom may evaluate **different** items (e.g., crowdsourcing)

# Fleiss' kappa

- Same fundamental idea of measuring the observed agreement compared to the agreement we would expect by chance.

$$\kappa = \frac{P_o - P_e}{1 - P_e}$$

- With N $>$ 2, we calculate agreement among **pairs** of annotators.

# Fleiss' Kappa

$n_{ij}$ is the number of annotators that agree on assigning the $i$-th class to the $j$-th item.

$o$ is the total number of annotators

K is the number of classes

For item i with n annotations, how many annotators agree, among all n(n-1) possible pairs.

$$P_i = \frac{1}{o(o-1)} \sum_{j=1}^{K} n_{ij}(n_{ij} - 1)$$

# Fleiss' Kappa

| [redacted] | Positive | Negative | Neutral | $P_i$ |
|---|---|---|---|---|
| Tweet 1 | 3 | 1 | 6 | **0.4** |
| Tweet 2 | 9 | 1 | 0 | |
| Tweet 3 | 3 | 5 | 2 | |
| Tweet 4 | 2 | 0 | 8 | |
| $p_j$ | | | | [redacted] |

$$P_1 = \frac{1}{10(10-1)} * (3*2 + 1*0 + 6*5) = \mathbf{0.4}$$

# Fleiss' Kappa

| [redacted] | Positive | Negative | Neutral | $P_i$ |
|---|---|---|---|---|
| Tweet 1 | 3 | 1 | 6 | 0.4 |
| Tweet 2 | 9 | 1 | 0 | 0.8 |
| Tweet 3 | 3 | 5 | 2 | 0.3111 |
| Tweet 4 | 2 | 0 | 8 | **0.6444** |
| $p_j$ | | | | [redacted] |

$$P_4 = \frac{1}{10(10-1)} * (2*1 + 0*-1 + 8*7) = \mathbf{0.6444}$$

# Fleiss' Kappa

| [redacted] | Positive | Negative | Neutral | $P_i$ |
|---|---|---|---|---|
| Tweet 1 | 3 | 1 | 6 | 0.4 |
| Tweet 2 | 9 | 1 | 0 | 0.8 |
| Tweet 3 | 3 | 5 | 2 | 0.3111 |
| Tweet 4 | 2 | 0 | 8 | 0.6444 |
| $p_j$ | | | | [redacted] |

N is the total number of items (Total Tweets in this example)

Average observed agreement among all items

$$P_o = \frac{1}{N}\sum_{i=1}^{N} P_i = \frac{1}{4} * (0.4 + 0.8 + 0.3111 + 0.6444) = 0.5388$$

# Fleiss' Kappa

| [redacted] | Positive | Negative | Neutral | $P_i$ |
|---|---|---|---|---|
| Tweet 1 | 3 | 1 | 6 | 0.4 |
| Tweet 2 | 9 | 1 | 0 | 0.8 |
| Tweet 3 | 3 | 5 | 2 | 0.3111 |
| Tweet 4 | 2 | 0 | 8 | 0.6444 |
| $p_j$ | **0.425** | | | [redacted] |

N is the total number of items (Total Tweets in this example)
$o$ is the total number of annotators
Probability of category j

$$p_j = \frac{1}{N * o} \sum_{i=1}^{N} n_{ij}$$

$$p_{positive} = \frac{1}{4 * 10} * (3 + 0 + 3 + 2) = \mathbf{0.425}$$

# Fleiss' Kappa

| [redacted] | Positive | Negative | Neutral | $P_i$ |
|---|---|---|---|---|
| Tweet 1 | 3 | 1 | 6 | 0.4 |
| Tweet 2 | 9 | 1 | 0 | 0.8 |
| Tweet 3 | 3 | 5 | 2 | 0.3111 |
| Tweet 4 | 2 | 0 | 8 | 0.6444 |
| $p_j$ | 0.425 | 0.175 | **0.4** | [redacted] |

N is the total number of items (Total Tweets in this example)
$o$ is the total number of annotators
Probability of category j

$$p_j = \frac{1}{N * o} \sum_{i=1}^{N} n_{ij}$$

$$p_{neutral} = \frac{1}{4 * 10} * (6 + 0 + 2 + 8) = \mathbf{0.4}$$

# Fleiss' Kappa

| [redacted] | Positive | Negative | Neutral | $P_i$ |
|---|---|---|---|---|
| Tweet 1 | 3 | 1 | 6 | 0.4 |
| Tweet 2 | 9 | 1 | 0 | 0.8 |
| Tweet 3 | 3 | 5 | 2 | 0.3111 |
| Tweet 4 | 2 | 0 | 8 | 0.6444 |
| $p_j$ | 0.425 | 0.175 | 0.4 | [redacted] |

Expected agreement by chance – joint probability two raters pick the same label is the product of their independent probabilities of picking that label
K is the number of classes

$$P_e = \sum_{j=1}^{K} p_j * p_j = 0.425 * 0.425 + 0.175 * 0.175 + 0.4 * 0.4 = \mathbf{0.3715}$$

# Fleiss' kappa

- Same fundamental idea of measuring the observed agreement compared to the agreement we would expect by chance.

$$\kappa = \frac{P_o - P_e}{1 - P_e} = \frac{0.5388 - 0.3715}{1 - 0.3715} = \mathbf{0.2662}$$

# Fleiss' Kappa

"Good" values are subject to interpretation, but rule of thumb

| Score Range | Interpretation |
|---|---|
| 0.81 - 1.00 | Almost Perfect |
| 0.61 - 0.80 | Substantial agreement |
| 0.41 - 0.60 | Moderate agreement |
| 0.21 - 0.40 | Fair agreement |
| 0.01 - 0.20 | Slight agreement |
| $< 0.0$ | Poor agreement |

# What about Ordinal/Regression Problems?

There are many agreement statistics. Find relevant research on the topic you are working on, then choose the staistic that is generally used.

- Krippendorff's alpha
- Pearson's r
- Kendalls' $\tau$
- Spearmans's $\rho$

# The End

The End