university of groningen

Quantitative Text Analysis – Essex Summer School

Supervised machine learning

dr. Martijn Schoonvelde

University of Groningen

Today's class

- An overview of supervised classification
- Lab session SML
- AOB week 1

A definition of supervised machine learning

"A form of machine learning where we aim to predict a variable, that, for at least part of our data, is known" (Van Atteveldt, 2022: p. 114)

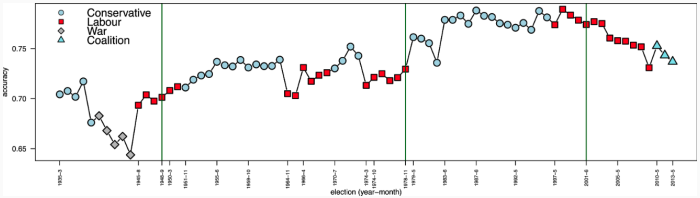
- We estimate a model based on some complete data, and then use the model to predict the expected outcome for some new cases, for which we do not know the outcome yet

In the context of QTA (supervised text analysis), we rely on textual data to make our predictions

- In the 'conventional' machine learning paradigm, the computer often requires large quantities of labelled data that map input features (often BoW representations) to output labels (e.g., sentiment, topics, hate speech, spam)
- But developments in this field are going fast; lots of work done to facilitate 'cheap learning' based on fewer examples (more on that next week)

Examples of text-based supervised machine learning

- Detecting Islamophobic hate speech on social media (Vidgen & Yasseri, 2020)
- Temporal focus of campaign communication (Müller, 2021)
- Measuring polarization in UK House of Commons (Peterson & Spirling, 2018)



Source: Peterson & Spirling, 2018

Workflow and terminology

1. Categorize a set of documents by hand – create a labeled dataset
2. Divide categorized documents in a training set and a test set
3. Use a supervised algorithm to 'learn' the relationship between the known labels and the features in the training set
4. Evaluate the classifier based on whether it correctly categorizes documents in the test set
5. Use the classifier to categorize unseen texts (not part of the training and the test set)

machine learning lingo	statistics lingo
feature	independent variable
label	dependent variable
labeled dataset	dataset with both independent and de
to train a model	to estimate
classifier (classification)	model to predict nominal outcomes
to annotate	to (manually) code (content analysis)

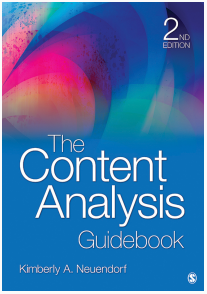
Source: Van Atteveldt *et al.* 2022

Creating a training set using hand coding

Why hand-code? To create a high-quality, labeled dataset for training and validating supervised machine learning models.

Steps (Grimmer, Roberts, and Stewart (2022):

1. **Define a codebook:** Clear definitions for each category or label.
2. **Select coders:** Choose trained or domain-knowledgeable coders.
3. **Sample documents:** Ensure diversity and balance in your selection (random sampling? purposive sampling?)
4. **Coordinate coding:** Give instructions, track progress, answer questions.
5. **Check reliability:** Use metrics like *Cohen's kappa*, *Krippendorff's alpha*.
6. **Address coder drift:** Hold periodic calibration sessions; re-code samples if needed.



Source: Neuendorf (2017), *The Content Analysis Guidebook*

Early example of supervised machine learning and text analysis

In the 1960s, statisticians Frederick Mosteller and David Wallace used a statistical approach to identify the authors of 12 disputed papers in the *Federalist Papers*.

- The authorship of these papers was disputed between Alexander Hamilton and James Madison.
- They concluded that Madison was the author of these papers.

This work represents an early application of supervised machine learning in authorship attribution or stylometry.

- Authorship attribution involves identifying the author of a text based on the semantic content and linguistic style of their writing.

Evaluating a Classifier: Accuracy

Imagine we have 100 Federalist Papers with known authorship (either Madison or Hamilton).

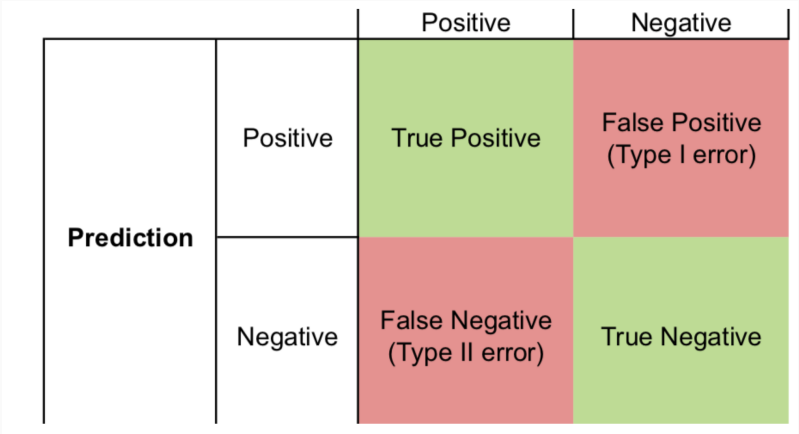
- 50 papers are written by Madison.
- 50 papers are written by Hamilton.

We aim to build a model to predict authorship.

- If we were to predict "Madison" for every paper, we would achieve 50% accuracy. While we're correct half the time, it is meaningless because it fails to identify any papers written by Hamilton.

Evaluating a classifier: confusion matrix

A confusion matrix is a table that summarizes the performance of a classifier.



Performance metrics

- **Precision**: This is the ratio of correctly predicted positive observations to the total predicted positives. High precision means that an algorithm returned more relevant results than irrelevant ones.
- **Recall**: This is the ratio of correctly predicted positive observations to all observations in the positive class. High recall means that an algorithm returned most of the relevant results.
- **F1 Score**: This is the weighted average of Precision and Recall. High F1 score means that both recall and precision are high.

Evaluating a classifier: confusion matrix

- Let's say we build a classifier and are able to predict 45 out of 50 Madison papers. But we also assign 5 papers incorrectly to Hamilton

		Reality	
		Madison	Hamilton
Prediction	Madison	45	7
	Hamilton	5	43

Evaluating a classifier: confusion matrix

		Reality	
		Madison	Hamilton
Prediction	Madison	45 (= **TP**)	7 (= **FP**)
	Hamilton	5 (= **FN**)	43 (= **TN**)

Evaluating a supervised classifier: precision and recall

Precision, recall and the F1 score are frequently used to assess classification performance:

- Precision: **TP / (TP + FP)**
  - *Do we identify only papers by Madison?*
- Recall: **TP / (TP + FN)**
  - *Do we identify all papers by Madison?*
- F1 score is a harmonic mean of precision and recall **2 * (Precision * Recall) / (Precision + Recall)**.

Evaluating supervised methods: precision and recall

- Precision: **TP / (TP + FP)**
  - $\rightarrow$ 45 / (45 + 7) = 0.865
- Recall: **TP / (TP + FN)**
  - $\rightarrow$ 45 / (45 + 5) = 0.9
- F1 score is a harmonic mean of precision and recall **2 * (Precision * Recall) / (Precision + Recall)**.
  - $\rightarrow$ 2*(0.90*0.865) / (0.90 + 0.865) = 0.882

How to find the 'best' model?

Precision, recall and F1 scores are useful, but it's up to us which classifier to choose. And as always, our decision will depends on our research goals.

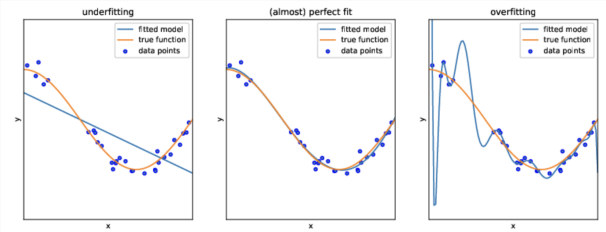
- If we are really determined not to miss any Madison papers, we may prioritise recall
- If we want to be certain that the papers we identify are indeed by Madison we may prioritise precision

F1 scores compromise between both but we don't want precision and recall to diverge too much

- For classifiers that also give us a prediction probabilities, we can also display a ROC curve, a plot that displays true positive against false positive at different probability thresholds

Generalisation and overfitting

- **Generalisation**: a classifier learns to correctly predict labels from given inputs not only in previously seen samples but also in previously unseen samples
- **Overfitting**: a classifier learns to correctly predict labels from given inputs in previously seen samples but fails to to do in in previously unseen samples

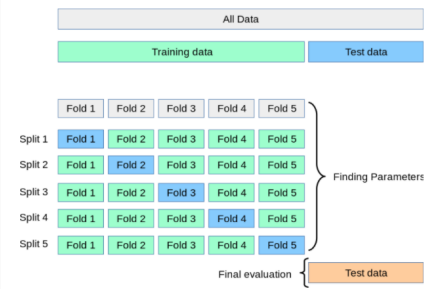


Source: Van Atteveldt, Trilling, Arcila Calderón, 2022

Cross validation

If we test the performance of multiple classifiers on our test data, we run the risk of overfitting our model to the test data

- One approach to address this issue is called k-fold cross validation where we separate our data into k separate training and test sets, and average our model results.
- If our classifier generalizes well, we should expect that our metric of choice is similar in all folds



Source `https://stats.stackexchange.com/`

Naïve Bayes classifier

- Use Bayes' rule to predict that a document has a certain label (e.g., positive or negative)
  - Also can be applied when we have more two labels (e.g., documents about sports, the economy or politics)
- Naïve assumption – all features are independent from each other, as if they are random draws from a vocabulary

Bayes' theorem:

$$P(label \mid features) = \frac{P(features \mid label)\, P(label)}{P(features)}$$

`NB: thanks to Stefan Müller for the examples in the next few slides`

Naïve Bayes classifier

Intuition: If we observe the term "fantastic" in a text, how likely is this text a positive review?

1. Determine frequency of positive and negative reviews (prior).
2. Assess probability of features given a particular class.
3. Get probability of a document belonging to each class (posterior).
4. Which posterior is highest?

Naïve Bayes classifier

Advantages

- Simple, fast, effective
- Relatively small training set required (if classes no too imbalanced). Easy to obtain probabilities

Disadvantages

- Assumption of conditional independence is problematic
  - In a document about football we expect words like ball, victory, league etc to cluster
- If feature is not in training set, it is disregarded for the classification

Naïve Bayes in quanteda

```
library(quanteda); library(quanteda.textmodels)
#get training set
dfmat_train <- tokens(c("positive bad negative horrible", "great fantastic nice")) %>% dfm()
class <- c("neg", "pos")

#train model
tmod_nb <- textmodel_nb(dfmat_train, class)

#get unlabelled test set
dfmat_test <- tokens(c("bad horrible afwul", "nice bad great", "great fantastic")) %>% dfm()

#predict class
predict(tmod_nb, dfmat_test, force = TRUE, type = "probability")

            neg       pos
text1 0.8573572 0.1426428
text2 0.2730748 0.7269252
text3 0.1712329 0.8287671
```

Other supervised classifiers

- Support Vector Machine, Logistic regression and other classifiers implemented in **quanteda.textmodels**
- Check out the book "Supervised Machine Learning for Text Analysis in R" by Emil Hvitfeldt and Julia Silge: `https://smltar.com/` for SML in the tidyverse
- Many, many tutorials online

"We will happily choose better features over better models every time" (Grimmer, Roberts, and Stewart, p. 185)

How to obtain high-quality labeled data?

Until recently:

- Researchers can recruit and train expert coders.
- Second, they can rely on crowd-workers on platforms such as Amazon Mechanical Turk (MTurk) (Benoit *et al.* 2016)

Since then some evidence has appeared that LLMs can do this as well:

- Gilardi et al. (2023) find that GPT-3 can accurately measure the topic of tweets, the stance or opinions of their authors, and frames included in them
- Ziems et al. (2023) find LLMs perform reasonably well in a wide range of datasets

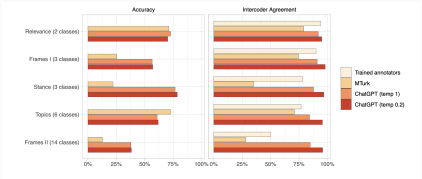


Figure 1: *ChatGPT zero-shot text annotation performance, compared to MTurk and trained annotators. ChatGPT's accuracy outperforms that of MTurk for four of the five tasks. ChatGPT's intercoder agreement outperforms that of both MTurk and trained annotators in all tasks. Accuracy means agreement with the trained annotators.*

Source Gilardi *et al.* 2023

How to obtain high-quality labeled data?

New developments allow for so-called 'cheap learning' techniques that might reduce data requirements without sacrificing performance or rigor

- weak supervision
- transfer learning
- prompt engineering

But we have to remain cognizant of the quality of our models: Error analysis – are our incorrect predictions systematic?