Predicting Authors

Olalekan Bello and Gaetano Dona-Jehan

May 11, 2021

Overview

Philosophy has an extensive history starting all the way from the era of the proto-philosophers such as Heraclitus and Thales to the early western philosophers like Plato and Aristotle to the later westerners such as Kant and Hume. Ideas are often transferred and refined from generation to generation and also within contemporaries. For this project, we aim to see if we could accurately distinguish ideas across a set of philosophers using both supervised and unsupervised machine learning tools. Our goal is to determine whether we can take a line from our set of authors and their works and accurately predict who wrote it.

We focus on four philosophers; Plato, Aristotle, Immanuel Kant and David Hume. We selected this group of philosophers for two reasons. One being that they are key figures in the history of philosophy. The second being that there exists relationships between the of authors that make the problem more interesting. Firstly, Plato and Aristotle are contemporaries as they both "published" works in the 4th century BC, whereas Kant and Hume published their essays in the 18th century. Given this, there might be a similarity that exists between authors in terms of writing styles and patterns that are common to their respective eras. There is also the issue of translations. Plato and Aristotle both originally wrote in Greek which has now been translated to English and Kant also originally wrote in German. Translations often contain the writing mannerisms of the translator, meaning that, although the original works are from two different authors, the translated works might share similar writing patterns due to being translated by the same translator.

In terms of relationships, there also exists a mix of ideas between the authors. Plato is know as the father of western philosophy which all the authors belong to and was actually a direct teacher of Aristotle. Kant and Hume were also "competitors" in a sense in that they belonged to different schools of thought within Western philosophy. Kant even famously credited Hume for "awaking him from his slumber".

Data and Models

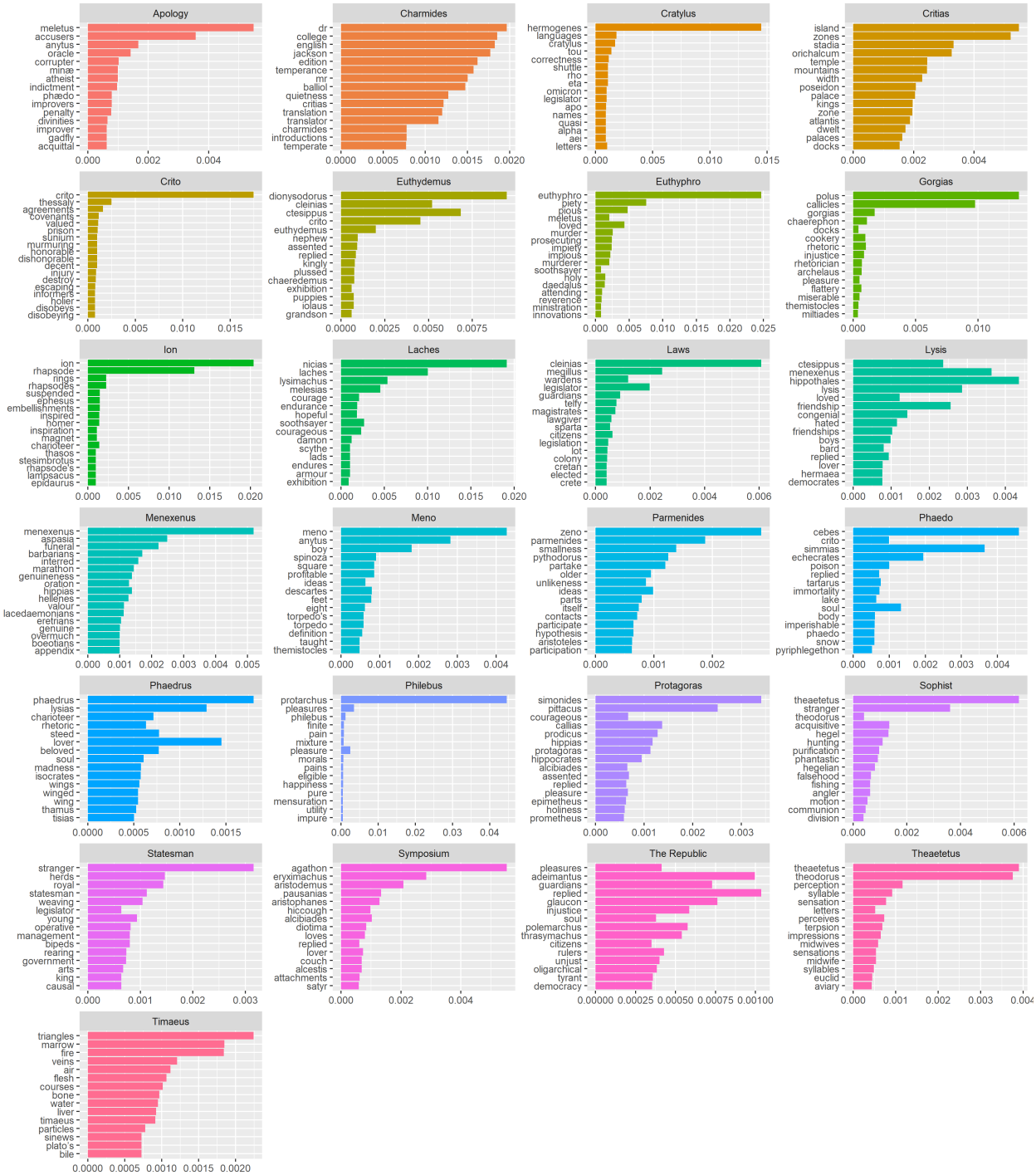
Our primary data source is the gutenberg project [1]. The gutenberg project is an online library of over 60,000 free e-books that span a long range of time and genres. We access this library using the "gutenbergr" package which allows us to directly download and process public domain works from the Gutenberg project collection as well as their metadata. The metadata that is given includes information about each work that we downloaded with their Gutenberg ID, title, information about the author, the language etc... We process this data using a number of text based packages in R. We break our text down into tokens whereby each token represents a feature of our dataset. Instead of using raw counts as our variables of interest, we use the slightly more sophisticated term-frequency inverse document frequency (tf_idf) [2]. Basically, the tf-idf is able to look within groups and identify the words that are most important to that particular group. This eliminates the need to manually remove stopwords such as "a, the, and, or" etc. because they would be weighted at 0 or very close to 0 because they are common across groups.

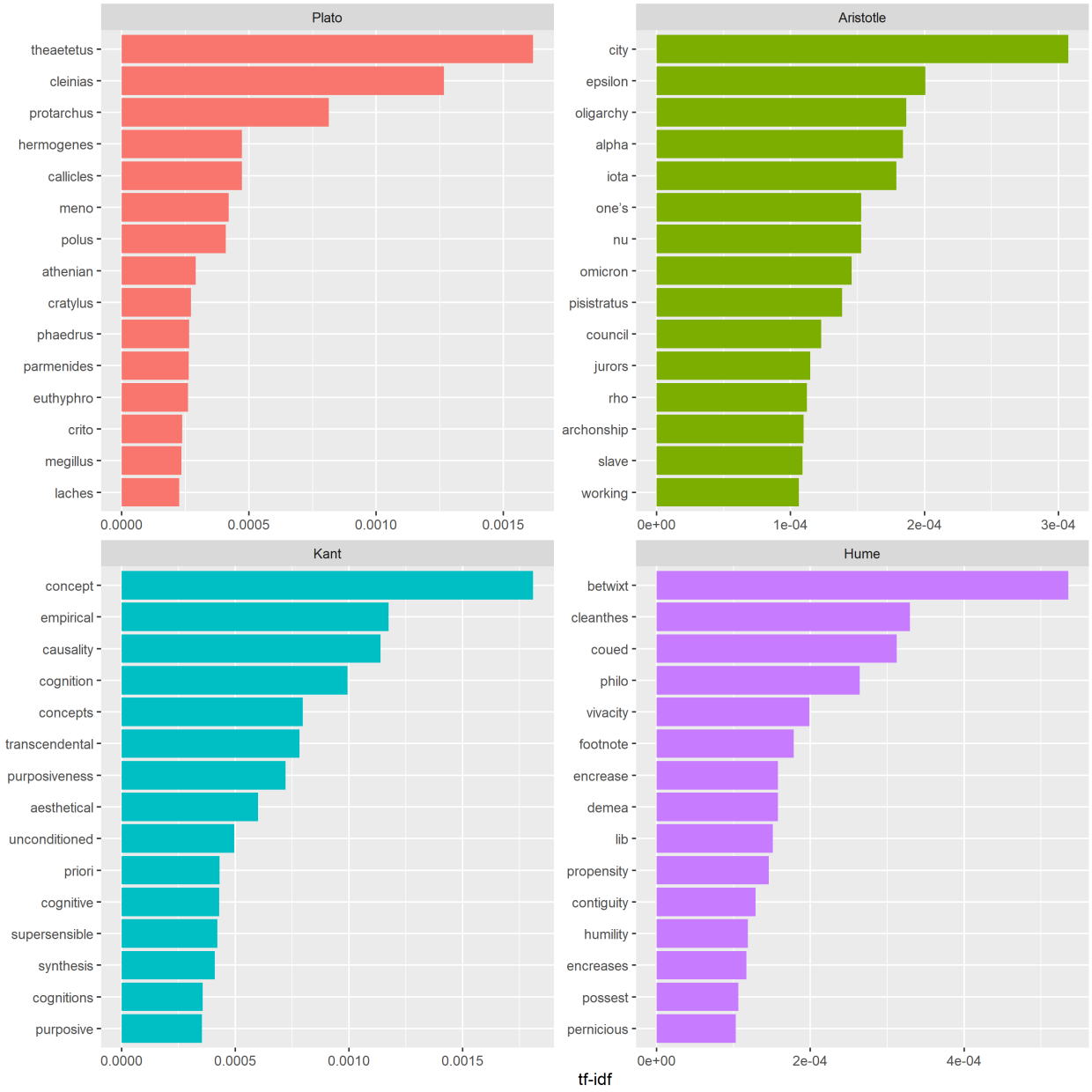
To start out, we take a look at the tf-idfs within Plato's works, we see that the words that most uniquely identify Plato's works are the names of characters which makes sense as Plato mostly expressed his philosophy through stories. Several of his works such as Euthyphro, Parmenides, Crito and Critias etc. are even named after the main character.

Next, we repeat the above step but this time across all authors. We see xyz

[1] https://www.gutenberg.org/

[2] Full list of packages used can be found in the Appendix


Apology
meletus
accusers
anytus
oracle
corrupter
minas
atheist
indictment
phaedo
improvers
penalty
divinities
improver
gadfly
acquittal
0.000
0.002
0.004
Charmides
dr
college
english
jackson
edition
temperance
mr
balliol
quietness
critias
translation
translator
charmides
introductions
temperate
0.0000
0.0005
0.0010
0.0015
0.0020
Cratylus
hermogenes
languages
cratylus
tou
correctness
shuttle
rho
eta
omicron
legislator
apo
names
quasi
alpha
aei
letters
0.000
0.005
0.010
0.015
Critias
island
zones
stadia
orichalcum
temple
mountains
width
poseidon
palace
kings
zone
atlantis
dwelt
palaces
docks
0.000
0.002
0.004
Crito
crito
thessaly
agreements
covenants
valued
prison
sunium
murmuring
honorable
dishonorable
decent
injury
destroy
escaping
informers
holier
disobeys
disobeying
0.000
0.005
0.010
0.015
Euthydemus
dionysodorus
cleinias
ctesippus
crito
euthydemus
nephew
assented
replied
kingly
plussed
chaeredemus
exhibition
puppies
iolaus
grandson
0.0000
0.0025
0.0050
0.0075
Euthyphro
euthyphro
piety
pious
meletus
loved
murder
prosecuting
impiety
impious
murderer
soothsayer
holy
daedalus
attending
reverence
ministration
innovations
0.000
0.005
0.010
0.015
0.020
0.025
Gorgias
polus
callicles
gorgias
chaerephon
docks
cookery
rhetoric
injustice
rhetorician
archelaus
pleasure
flattery
miserable
themistocles
miltiades
0.000
0.005
0.010
Ion
ion
rhapsode
rings
rhapsodes
suspended
ephesus
embellishments
inspired
homer
inspiration
magnet
charioteer
thasos
stesimbrotus
rhapsode's
lampsacus
epidaurus
0.000
0.005
0.010
0.015
0.020
Laches
nicias
laches
lysimachus
melesias
courage
endurance
hopeful
soothsayer
courageous
damon
scythe
lads
endures
armour
exhibition
0.000
0.005
0.010
0.015
0.020
Laws
cleinias
megillus
wardens
legislator
guardians
telfy
magistrates
lawgiver
sparta
citizens
legislation
lot
colony
cretan
elected
crete
0.000
0.002
0.004
0.006
Lysis
ctesippus
menexenus
hippothales
lysis
loved
friendship
congenial
hated
friendships
boys
bard
replied
lover
hermaea
democrates
0.000
0.001
0.002
0.003
0.004
Menexenus
menexenus
aspasia
funeral
barbarians
interred
marathon
genuineness
oration
hippias
hellenes
valour
lacedaemonians
eretrians
genuine
overmuch
boeotians
appendix
0.000
0.001
0.002
0.003
0.004
0.005
Meno
meno
anytus
boy
spinoza
square
profitable
ideas
descartes
feet
eight
torpedo's
torpedo
definition
taught
themistocles
0.000
0.001
0.002
0.003
0.004
Parmenides
zeno
parmenides
smallness
pythodorus
partake
older
unlikeness
ideas
parts
itself
contacts
participate
hypothesis
aristoteles
participation
0.000
0.001
0.002
Phaedo
cebes
crito
simmias
echecrates
poison
replied
tartarus
immortality
lake
soul
body
imperishable
phaedo
snow
pyriphlegethon
0.000
0.001
0.002
0.003
0.004
Phaedrus
phaedrus
lysias
charioteer
rhetoric
steed
lover
beloved
soul
madness
isocrates
wings
winged
wing
thamus
tisias
0.0000
0.0005
0.0010
0.0015
Philebus
protarchus
pleasures
philebus
finite
pain
mixture
pleasure
morals
pains
eligible
happiness
pure
mensuration
utility
impure
0.00
0.01
0.02
0.03
0.04
Protagoras
simonides
pittacus
courageous
callias
prodicus
hippias
protagoras
hippocrates
alcibiades
assented
replied
pleasure
epimetheus
holiness
prometheus
0.000
0.001
0.002
0.003
Sophist
theaetetus
stranger
theodorus
acquisitive
hegel
hunting
purification
phantastic
hegelian
falsehood
fishing
angler
motion
communion
division
0.000
0.002
0.004
0.006
Statesman
stranger
herds
royal
statesman
weaving
legislator
young
operative
management
bipeds
rearing
government
arts
king
causal
0.000
0.001
0.002
0.003
Symposium
agathon
eryximachus
aristodemus
pausanias
aristophanes
hiccough
alcibiades
diotima
loves
replied
lover
couch
alcestis
attachments
satyr
0.000
0.002
0.004
The Republic
pleasures
adeimantus
guardians
replied
glaucon
injustice
soul
polemarchus
thrasymachus
citizens
rulers
unjust
oligarchical
tyrant
democracy
0.00000
0.00025
0.00050
0.00075
0.00100
Theaetetus
theaetetus
theodorus
perception
syllable
sensation
letters
perceives
terpsion
impressions
midwives
sensations
midwife
syllables
euclid
aviary
0.000
0.001
0.002
0.003
0.004
Timaeus
triangles
marrow
fire
veins
air
flesh
courses
bone
water
liver
timaeus
particles
sinews
plato's
bile
0.0000
0.0005
0.0010
0.0015
0.0020
tf-idf



Supervised learning model

For our supervised learning model, we employ the use of a multinomial regression model with lasso regularization using the glmnet package in r. The tokens and their tf-idfs represent the features of our data. This method is well suited for text prediction as given the very high dimensional space of our features, the lasso regularization allows for the penalization of features (down to zero at times) and so the final model will only include features (tokens) that the algorithm considers to be key to prediction [3].

[3] 3

First, we split our data into a training and test set and examine the raw counts of lines by author. From the table below, we see that we do not have balance as Plato is overly represented in our training sample. To address this, authors with the majority of the lines are down-sampled after our text-preprocessing steps in order to achieve a balanced sample of authors to potentially improve our prediction. We also use a max of 1600 tokens [4] We explore different values for our penalty hyper-parameter and we use the one that returns the best results in terms of prediction accuracy as our final model. In dealing with the random variation in our train/test splits, we use k-fold cross validation with 10 folds.

```
## # A tibble: 4 x 2
##       n author
##   <int> <chr>
## 1 93319 Plato
## 2 41364 Kant
## 3 31689 Hume
## 4 23183 Aristotle
```

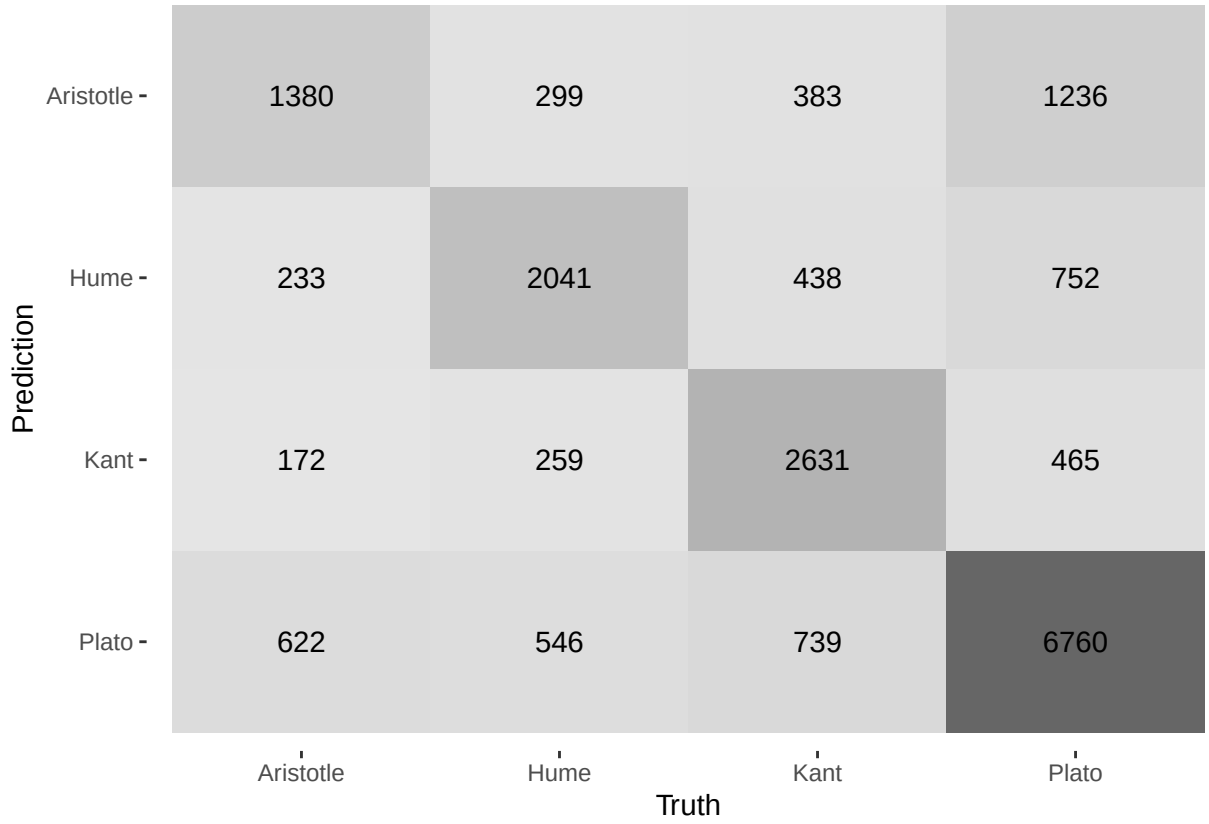
The table below reports the "best" model across a range of penalties. We see that our best model has an accuracy of about 67 percent. This is not "awesome" in terms of accuracy but is fairly reasonable. We are attempting to predict the author line by line within the same subject area of philosophy without any other predictors other than the text itself which is a fairly challenging task. There are also the aforementioned challenges of the connections that exist between the authors making the classification task even more difficult.

```
## # A tibble: 5 x 7
##     penalty .metric  .estimator  mean     n  std_err .config
##       <dbl> <chr>    <chr>      <dbl> <int>    <dbl> <chr>
## 1 0.000464  accuracy multiclass 0.673    10 0.000977 Preprocessor1_Model04
## 2 0.000129  accuracy multiclass 0.673    10 0.00100  Preprocessor1_Model03
## 3 0.0000359 accuracy multiclass 0.673    10 0.00113  Preprocessor1_Model02
## 4 0.00001   accuracy multiclass 0.673    10 0.00117  Preprocessor1_Model01
## 5 0.00167   accuracy multiclass 0.664    10 0.00121  Preprocessor1_Model05
```

The confusion matrix below for the first fold shows exactly how the model classifies and mis-classifies. We see that the diagonal is well populated which is good. The model seems to do best at classifying Plato.

[4] 4

This makes sense because as previously discussed, Plato uses the name of characters a lot in his works that would not be used by another author making it easier to identify that a line was written by him. With misclassification, we see that Plato is often mistaken for Aristotle which again makes is reasonable as Plato was Aristotle's teacher and so naturally there would be a transfer of ideas between their works.



Lastly,we evaluate our model on the test set. We have an accuracy of 66.7% which is very close to our in-sample accuracy.

```
## # A tibble: 2 x 4
##   .metric  .estimator .estimate .config
##   <chr>    <chr>          <dbl> <chr>
## 1 accuracy multiclass     0.667 Preprocessor1_Model1
## 2 roc_auc  hand_till      0.860 Preprocessor1_Model1
```

Let's check and see which lines the model seems to do best at in the test set.

Best predicted lines

```
## # A tibble: 8 x 4
```

```
##   author    text                                 title                       value
##   <chr>     <chr>                                <chr>                       <dbl>
## 1 Aristotle Self-Control.                        The Ethics of Aristotle      1.00
## 2 Aristotle Choice.                              The Ethics of Aristotle      1.00
## 3 Hume      fancy.                               A Treatise of Human Nature   1.00
## 4 Hume      considerable resemblance?            Dialogues Concerning Natu~   1.00
## 5 Kant      conditioned.                         The Critique of Pure Reas~   1.00
## 6 Kant      Judgement.                           Kant's Critique of Judgem~   1.00
## 7 Plato     CEPHALUS - SOCRATES - POLEMARCHUS    The Republic                 1.00
## 8 Plato     SOCRATES - CLEITOPHON - POLEMARCHU~ The Republic                 1.00
```

The lines are pretty short and most are even single words but the predictions make sense. As we see with Plato, lines with the character names are almost guaranteed to be written by him. Self-control is also a big theme within Aristotle work's as he was the one who heavily developed the idea of the golden-mean.

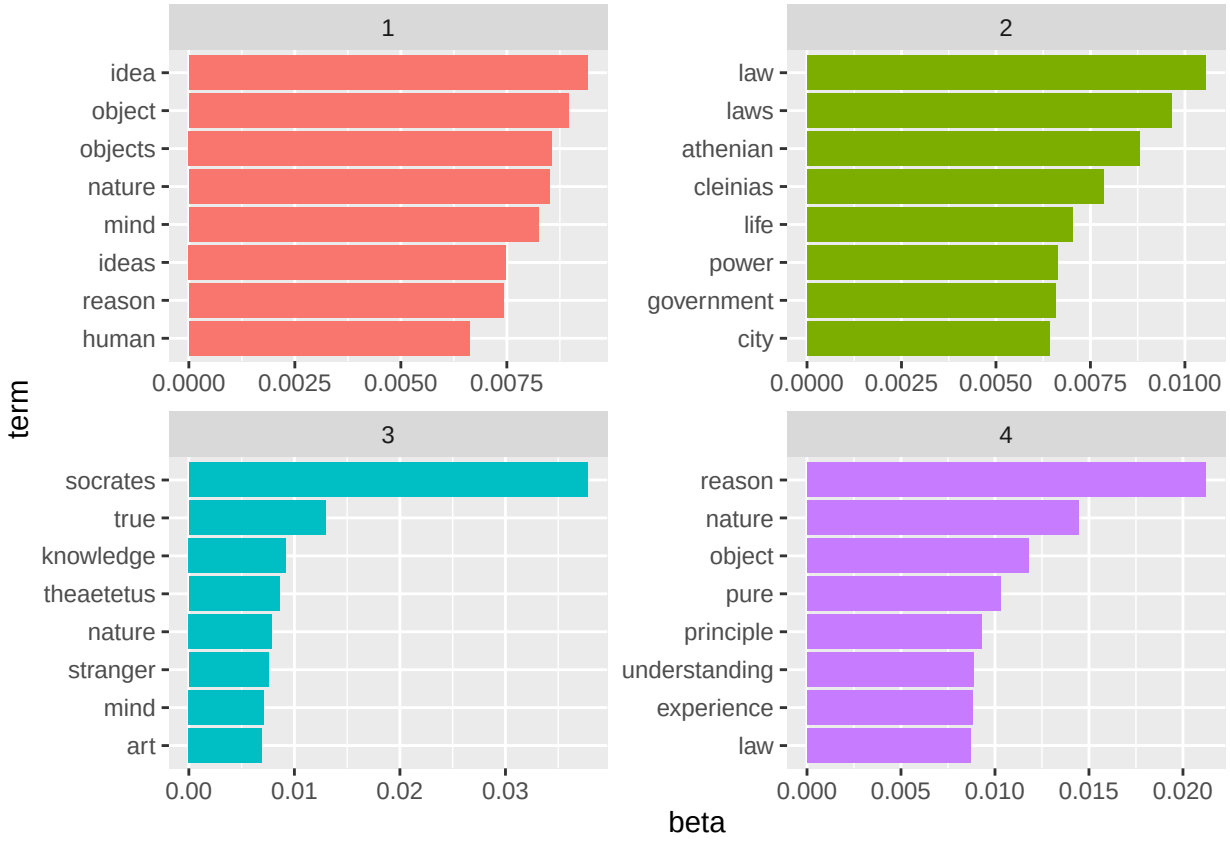
Unsupervised learning model

One of the most used models in text mining to analyze series of unlabeled, or even labelled, documents is topic models. Topic modeling is an unsupervised machine learning that "scans" these documents and finds natural cluster of words or phrases called topics. For example, say that we have two documents: an article about cars and an article about groceries. The common words in groceries would be "carrots", "meat" and "milk" whereas the common words in cars would be "gas", "horsepower" and "exhaust". Consequently, each of these words would be respectively grouped together by the model creating two clusters. This might not seem useful when we only consider two documents, but it becomes extremely useful when we have hundreds or more documents and we want to cluster them broadly on areas of interest. In fact, this process is most likely used by major news networks or social media outlets. For example, say that someone logs in to their New York Times (NYT) account and primarily reads articles related to international politics or economics. The NYT will then use this data to recommend other articles that have the international politics or economics topic. However, the question remains on how to create such a model.

The main approach that is used the Latent Dirichlet Allocation (LDA) method. This method assumes two things: similar topics use similar words and documents are a mixture of several topics. In the example above with the grocery and car articles, there are shared words between the two topics such as "budget", "price" and "environment". The advantage of using the LDA is that the documents that are studied can "overlap"

with each other in terms of word or phrase usage which reflects the use of natural language. Since we are dealing with philosophical works, we decided to use the LDA as we want to allow for the possibility that the philosophers might use the same words or phrases in their works as the others. The data preparation process is very similar to the supervised model, in which we break down out texts into token representing a particular word. The one difference that we make in the unsupervised learning model is that we are removing any words that appear less than 50 times in our tokenized dataset. The reason is that we want to see how well the model performs when there is a significant amount of overlap between the various works. This is because we believe that the words that are used less than 50 times are unique to a specific author, thus creating a clear distinction between them and the other authors.

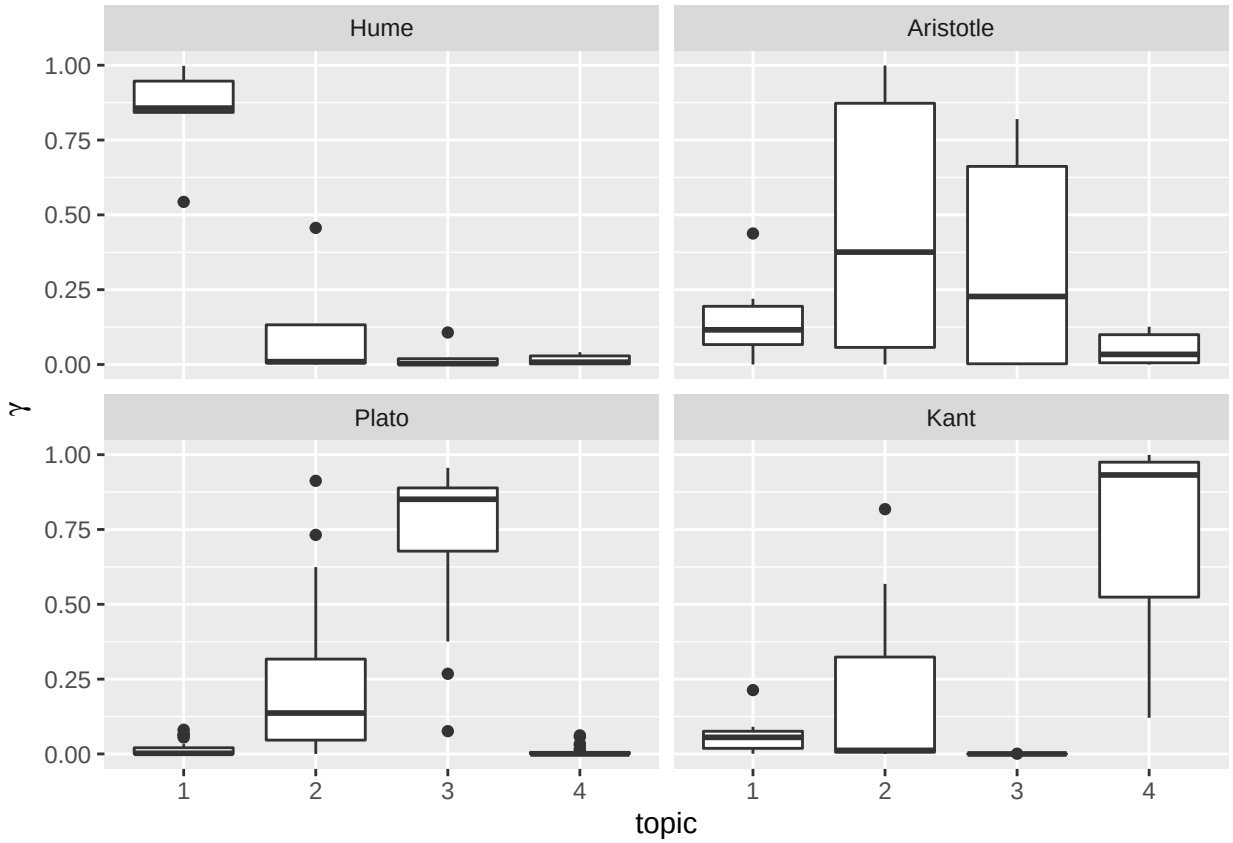
To perform the LDA method in R, we are using the LDA command in the topicmodels package. This command highlights one of the key "issues" of topic modeling: the need to specify a number of topics. In the case of major news networks, one can see the various topics that describe an article (health, business, politics, etc...) that have been decided by these networks. The topic models do not inherently give names to the topics they simply cluster the documents based on the number of cluster points, i.e. topics, specified. However, since we know that our documents have been written by four authors, we specify to our model that there are four topics to cluster around.



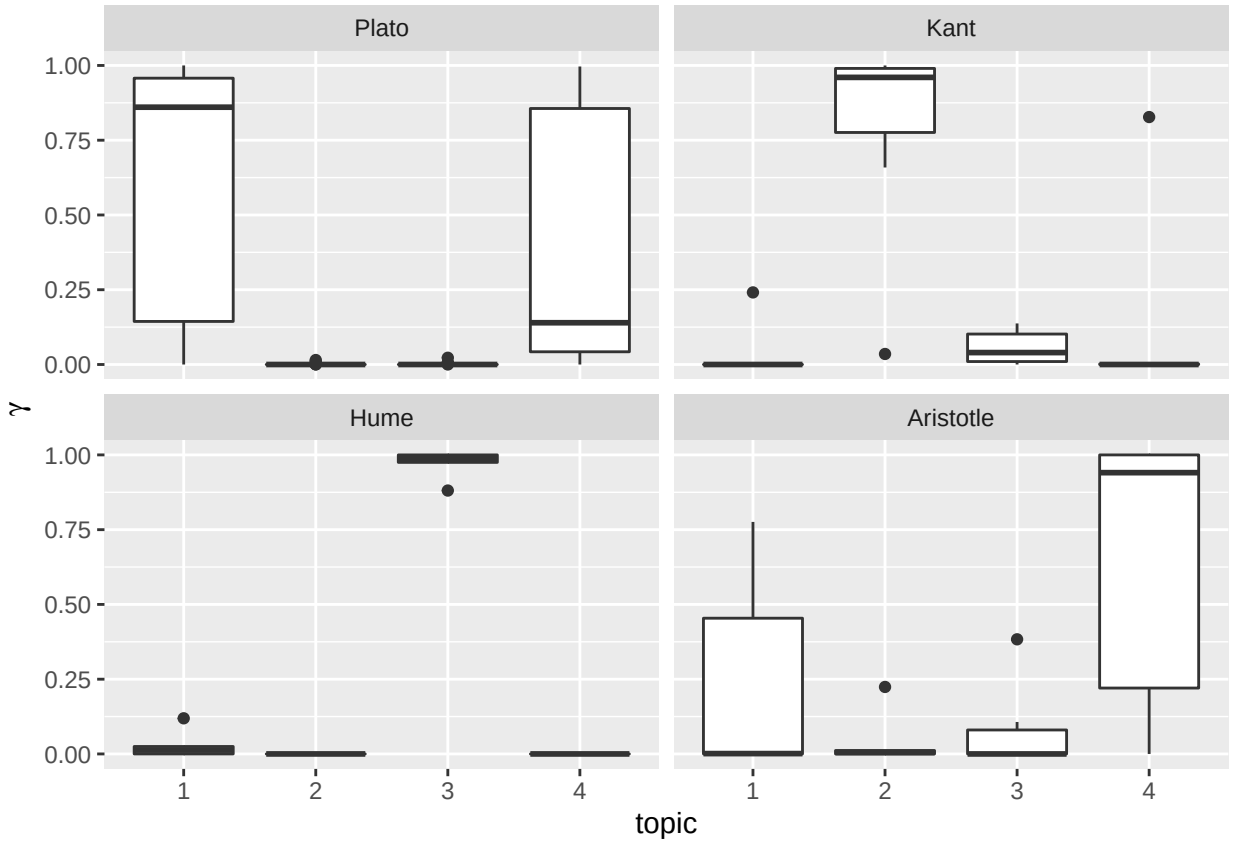
As we can see in the graph above, the issue of needing to specify the number and the name of topics is clear. The graph allows us to infer which topic belongs to which author, but it does not directly give us the names of the authors associated with each topic. We can conclude that topic 2 and 3 are associated with the Greek Philosophers due to the words Athenian, Cleinias, Socrates and Theaetetus, therefore leaving topics 1 and 4 to the more modern philosophers Hume and Kant. However, we are unable to confidently say which topic is specifically associated to a particular author. From term-frequency inverse document frequency within Plato's works, the words that most uniquely identify Plato's works are name of characters. Since topic 3 has two names that are mentioned, and that are in the top four most frequent words, we can conclude that topic 3 refers to Plato, and topic 2 to Aristotle. We believe that topic 1 represents Hume because we are somewhat familiar with Kant's philosophy about nature and principles, words that are found in topic 4.

```
## # A tibble: 172 x 4
##    Author    Title                                          topic     gamma
##    <chr>     <chr>                                          <int>     <dbl>
##  1 Aristotle Aristotle on the art of poetry                     1 0.112
##  2 Aristotle Politics: A Treatise on Government                 1 0.0513
##  3 Aristotle The Athenian Constitution                          1 0.0000304
##  4 Aristotle The Categories                                     1 0.438
##  5 Aristotle The Ethics of Aristotle                            1 0.220
##  6 Aristotle The Poetics of Aristotle                           1 0.119
##  7 Hume      A Treatise of Human Nature                         1 0.998
##  8 Hume      An Enquiry Concerning Human Understanding          1 0.947
##  9 Hume      An Enquiry Concerning the Principles of Morals     1 0.857
## 10 Hume      Dialogues Concerning Natural Religion              1 0.842
## # ... with 162 more rows
```

However, this is not enough. We are interested in seeing the document per topic probability meaning that we want to see what is the probability that a document is situated in a particular topic. For example, Aristotle's book "Aristotle on the Art of Poetry" has an 11% chance of coming from topic 1, which we assume to be Hume. In order to get a better idea, we look at the boxplot that represents the probability, denoted as γ, that the author is in the respective topic.



Unfortunately, our model does not have a very good predictive capability, although it is important to note that we removed words that were used less than 50 times, which would affect the results. However, considering this limitation, the model does a relatively good job at distinguishing works written by Hume and Kant from work written in a different era, written by Aristotle and Plato. Furthermore, Hume's works, topic 1, have almost been perfectly associated with being written by Hume. Kant's association with topic 4 is also high. However, the model is terrible at distinguishing works between Plato and Aristotle. Both authors have written in the same language and have both been translated. As mentioned earlier, translations often contain the writing mannerisms of the translator, meaning that translated works might share similar writing patterns as they might be translated by the same translator. In fact, Plato's works have been mostly attributed to Aristotle as an author, as shown by the table in the appendix, possibly because Aristotle was Plato's student. This seems to be supported by the figure below where we reran LDA model without exlcuding the words that have appeared 50 times or less. As we can see, both the prediction accuracy for Hume and Kant have significantly increased.



Conclusion

Appendix

```
## # A tibble: 30 x 5
##    Author   Title                                              topic gamma consensus
##    <chr>    <chr>                                              <int> <dbl> <chr>
##  1 Aristot~ "Aristotle on the art of poetry"                       3 0.734 Plato
##  2 Aristot~ "The Categories"                                       3 0.445 Plato
##  3 Aristot~ "The Poetics of Aristotle"                             3 0.820 Plato
##  4 Kant     "Of the Injustice of Counterfeiting Books\r\n~         2 0.818 Aristotle
##  5 Kant     "Perpetual Peace\nA Philosophical Essay"               2 0.568 Aristotle
##  6 Plato    "Apology"                                              3 0.501 Aristotle
##  7 Plato    "Charmides"                                            3 0.902 Aristotle
##  8 Plato    "Cratylus"                                             3 0.946 Aristotle
```

```
##  9 Plato    "Critias"                                          2 0.732 Aristotle
## 10 Plato    "Crito"                                            3 0.561 Aristotle
## # ... with 20 more rows
```

Packages used not described in the main text

tidytext tokenizer ggplot2 stringr