

Detecting Hate speech using Natural Language processing

Vinita V. Vader¹

¹ University of Oregon

Author Note

Vinita V.Vader, University of Oregon, Department of Psychology, 1227 University of Oregon, Eugene, OR, 97403

Correspondence concerning this article should be addressed to Vinita V. Vader, Department of Psychology, 1227 University of Oregon, Eugene, OR, 97403. E-mail: vvader@uoregon.edu

Detecting Hate speech using Natural Language processing

Research problem

Describe the task you want to achieve. What is the outcome of interest? What are you trying to predict? Why is it important? What are the potential benefits of having a predictive model for this outcome? Discuss potential applications of such a model.

The lines between freedom of speech and offensive speech are filled with confusion in the age of freedom of choice and information dissemination on social media platforms. Honesty in thought could be a potential violation of someone's right to function in a respectable manner in a society. The constant suspension of morality in the milieu of right to perspectives has been detrimental to attaining a universal grammar for understanding the meaning of rights and violation of rights. Although the philosophical debate around this issue could surface several deep seated issues about morality of offense and integrity, it is important to consider that discomfort caused to anyone is crucial to any form of action - online or offline - to be considered as harmful for the society. A recent study by (Williams, Burnap, Javed, Liu, & Ozalp, 2020) demonstrated how hate speech could influence political votes, terror attacks and promote criminal behavior by normalizing violence.

In the light of this discussion, hate speech behaviors on social media platforms become vital to studying the precedents and possibly, conveyers of hate crimes. This report focuses on offensive language detection, specifically, hate speech detection in twitter data. Hate speech is defined as "language that is used to express hatred towards a targeted group or is intended to be derogatory, to humiliate, or to insult the members of the group," (Davidson, Warmesley, Macy, & Weber, 2017). Natural Language processing (NLP) has gained prominence in understanding the qualitative information in human language. The development of context-dependent word embeddings such as Bidirectional Encoder

Representations from Transformers or BERT the next state-of-art work in NLP which outperformed previous models. Among the competing BERT models that emerged through the corpus of text data available to for-profit and non-profit companies, Facebook's RoBERTa model has gained attention with its several advantages due to the training data it is based on. Ten-times larger dataset, longer training, increased batch size, excluding next sentence predicting task, using byte-level encoding with bigger vocabulary and dynamic masking pattern changing are some of the improvements that this model has been trained with.

This report explores experiments in classifying hate and no hate speech using the word embedding from pre-trained RoBERTa model. Different layers emerging from the RoBERTa based model will be utilized to explore varying number of dimensions as predictors of classifying text obtained from Twitter as hate and no hate speech. Comparisons will be made across classification models such as Logistic regression with regularization and random forest.

Bias in model performance, assessed based on sensitivity and accuracy analyses, will provide implications about groupings in the real world. Given that these algorithms might be used in legal and political settings (eg., what is the probability that a certain tweet invoked violence against group X), it is important to assess their performance in generating appropriate inferences. A biased algorithm may lead to injustice towards certain groups in the society. The models resulting through the analysis in this report could be used as a starting point for evaluating bias in algorithms classifying hate speech. The results could also provide guidelines for caution while trusting outcomes based on RoBERTa models.

Description of the data

Describe core features of the data, any additional features you produced from existing features and how, basic descriptive statistics about these features, and

any missing data analysis you conduct. The description should be sufficiently clear that the instructor understands all the variables included in your modeling.

```
## $Continuous
```

```
## # A tibble: 1,000 x 0
```

```
##
```

```
## $Categorical
```

```
##      label var_type      n missing_n missing_percent levels_n levels
```

```
## text    text    <chr> 1000          0           0.0      1000      -
```

```
## label label    <chr> 1000          0           0.0         2      -
```

```
##      levels_count levels_percent
```

```
## text              -              -
```

```
## label              -              -
```

Data generation process

The dataset has been built synthetically as a result of an AI improvement initiative by the Facebook AI division. The process of generating this data has been described below: Writers recruited for the task were provided with a certain context and target for whom they created some speech text categorized as hate or no hate (H and NH) speech. The target of the speech and the sentiment - H or NH - were set *a priori* to the generation of speech text by each writer. These texts were inputs for models (generated by Facebook) to test algorithms which best categorized the text the speeches as H or NH. If the model classified the text correctly, the speech text is retained as a good training example. If the model failed to classify the text correctly a verifier (person) was called in. The verifier - if he disagreed with the writer, the speech was trashed but if he agreed with the writer the speech text was retained for testing or developing the model.

Data features

The original data consists of 40623 synthetically produced speech texts. For the current project we will be using 1000 randomly sampled speech texts. The original data file consists of the following columns - .

For the purpose of the current study we are only interested in two columns, namely - . The text variable consists of all the synthetically generated speech texts. The label variable consists of the type of text speech produced. This is the *a priori* hyp.

No NA's or missing values were found in the data. Total number of texts labeled as hate speech were c("hate," "nothate") and those labeled as non-hate speech were c(562, 438) 1.

Description of the models

List at least three different modeling approaches you apply to this dataset.

Describe each model, why the given model was selected, which hyperparameters to be optimized and how. Also, discuss how you plan to evaluate model performance.

For classifying the text in the given data generalized linear models will be applied. The data will be modeled using Logistic regression without regularization and with regularization (ridge and lasso). For optimization of models, the λ hyperparameter was tuned for ridge regression and lasso regression. The α hyperparameter was set to 0 for Ridge regularization and 1 for Lasso regularization.

The performance evaluation metrics on the test data would include computing *LogLikelihood (LL)*, *Area under the curve (AUC)*, *Accuracy (ACC)*, *True Positive rate (TPR)*, *True Negative rate (TNR)*, *False Positive rate (FPR)*, and *Precision (PRE)*. #

108 Models: Roberta - layer 12 Logistic regression Logistic regression Lasso Logistic regression

109 Ridge

110 Roberta - layer 7 Logistic regression Logistic regression Lasso Logistic regression

111 Ridge

112	##	[1]	"text"	"label"	"id_col"	"Dim1"	"Dim2"	"Dim3"	"Dim4"	"Dim5"
113	##	[9]	"Dim6"	"Dim7"	"Dim8"	"Dim9"	"Dim10"	"Dim11"	"Dim12"	"Dim13"
114	##	[17]	"Dim14"	"Dim15"	"Dim16"	"Dim17"	"Dim18"	"Dim19"	"Dim20"	"Dim21"
115	##	[25]	"Dim22"	"Dim23"	"Dim24"	"Dim25"	"Dim26"	"Dim27"	"Dim28"	"Dim29"
116	##	[33]	"Dim30"	"Dim31"	"Dim32"	"Dim33"	"Dim34"	"Dim35"	"Dim36"	"Dim37"
117	##	[41]	"Dim38"	"Dim39"	"Dim40"	"Dim41"	"Dim42"	"Dim43"	"Dim44"	"Dim45"
118	##	[49]	"Dim46"	"Dim47"	"Dim48"	"Dim49"	"Dim50"	"Dim51"	"Dim52"	"Dim53"
119	##	[57]	"Dim54"	"Dim55"	"Dim56"	"Dim57"	"Dim58"	"Dim59"	"Dim60"	"Dim61"
120	##	[65]	"Dim62"	"Dim63"	"Dim64"	"Dim65"	"Dim66"	"Dim67"	"Dim68"	"Dim69"
121	##	[73]	"Dim70"	"Dim71"	"Dim72"	"Dim73"	"Dim74"	"Dim75"	"Dim76"	"Dim77"
122	##	[81]	"Dim78"	"Dim79"	"Dim80"	"Dim81"	"Dim82"	"Dim83"	"Dim84"	"Dim85"
123	##	[89]	"Dim86"	"Dim87"	"Dim88"	"Dim89"	"Dim90"	"Dim91"	"Dim92"	"Dim93"
124	##	[97]	"Dim94"	"Dim95"	"Dim96"	"Dim97"	"Dim98"	"Dim99"	"Dim100"	"Dim101"
125	##	[105]	"Dim102"	"Dim103"	"Dim104"	"Dim105"	"Dim106"	"Dim107"	"Dim108"	"Dim109"
126	##	[113]	"Dim110"	"Dim111"	"Dim112"	"Dim113"	"Dim114"	"Dim115"	"Dim116"	"Dim117"
127	##	[121]	"Dim118"	"Dim119"	"Dim120"	"Dim121"	"Dim122"	"Dim123"	"Dim124"	"Dim125"
128	##	[129]	"Dim126"	"Dim127"	"Dim128"	"Dim129"	"Dim130"	"Dim131"	"Dim132"	"Dim133"
129	##	[137]	"Dim134"	"Dim135"	"Dim136"	"Dim137"	"Dim138"	"Dim139"	"Dim140"	"Dim141"
130	##	[145]	"Dim142"	"Dim143"	"Dim144"	"Dim145"	"Dim146"	"Dim147"	"Dim148"	"Dim149"
131	##	[153]	"Dim150"	"Dim151"	"Dim152"	"Dim153"	"Dim154"	"Dim155"	"Dim156"	"Dim157"
132	##	[161]	"Dim158"	"Dim159"	"Dim160"	"Dim161"	"Dim162"	"Dim163"	"Dim164"	"Dim165"
133	##	[169]	"Dim166"	"Dim167"	"Dim168"	"Dim169"	"Dim170"	"Dim171"	"Dim172"	"Dim173"

134 ## [177] "Dim174" "Dim175" "Dim176" "Dim177" "Dim178" "Dim179" "Dim180" "Dim181"
135 ## [185] "Dim182" "Dim183" "Dim184" "Dim185" "Dim186" "Dim187" "Dim188" "Dim189"
136 ## [193] "Dim190" "Dim191" "Dim192" "Dim193" "Dim194" "Dim195" "Dim196" "Dim197"
137 ## [201] "Dim198" "Dim199" "Dim200" "Dim201" "Dim202" "Dim203" "Dim204" "Dim205"
138 ## [209] "Dim206" "Dim207" "Dim208" "Dim209" "Dim210" "Dim211" "Dim212" "Dim213"
139 ## [217] "Dim214" "Dim215" "Dim216" "Dim217" "Dim218" "Dim219" "Dim220" "Dim221"
140 ## [225] "Dim222" "Dim223" "Dim224" "Dim225" "Dim226" "Dim227" "Dim228" "Dim229"
141 ## [233] "Dim230" "Dim231" "Dim232" "Dim233" "Dim234" "Dim235" "Dim236" "Dim237"
142 ## [241] "Dim238" "Dim239" "Dim240" "Dim241" "Dim242" "Dim243" "Dim244" "Dim245"
143 ## [249] "Dim246" "Dim247" "Dim248" "Dim249" "Dim250" "Dim251" "Dim252" "Dim253"
144 ## [257] "Dim254" "Dim255" "Dim256" "Dim257" "Dim258" "Dim259" "Dim260" "Dim261"
145 ## [265] "Dim262" "Dim263" "Dim264" "Dim265" "Dim266" "Dim267" "Dim268" "Dim269"
146 ## [273] "Dim270" "Dim271" "Dim272" "Dim273" "Dim274" "Dim275" "Dim276" "Dim277"
147 ## [281] "Dim278" "Dim279" "Dim280" "Dim281" "Dim282" "Dim283" "Dim284" "Dim285"
148 ## [289] "Dim286" "Dim287" "Dim288" "Dim289" "Dim290" "Dim291" "Dim292" "Dim293"
149 ## [297] "Dim294" "Dim295" "Dim296" "Dim297" "Dim298" "Dim299" "Dim300" "Dim301"
150 ## [305] "Dim302" "Dim303" "Dim304" "Dim305" "Dim306" "Dim307" "Dim308" "Dim309"
151 ## [313] "Dim310" "Dim311" "Dim312" "Dim313" "Dim314" "Dim315" "Dim316" "Dim317"
152 ## [321] "Dim318" "Dim319" "Dim320" "Dim321" "Dim322" "Dim323" "Dim324" "Dim325"
153 ## [329] "Dim326" "Dim327" "Dim328" "Dim329" "Dim330" "Dim331" "Dim332" "Dim333"
154 ## [337] "Dim334" "Dim335" "Dim336" "Dim337" "Dim338" "Dim339" "Dim340" "Dim341"
155 ## [345] "Dim342" "Dim343" "Dim344" "Dim345" "Dim346" "Dim347" "Dim348" "Dim349"
156 ## [353] "Dim350" "Dim351" "Dim352" "Dim353" "Dim354" "Dim355" "Dim356" "Dim357"
157 ## [361] "Dim358" "Dim359" "Dim360" "Dim361" "Dim362" "Dim363" "Dim364" "Dim365"
158 ## [369] "Dim366" "Dim367" "Dim368" "Dim369" "Dim370" "Dim371" "Dim372" "Dim373"
159 ## [377] "Dim374" "Dim375" "Dim376" "Dim377" "Dim378" "Dim379" "Dim380" "Dim381"
160 ## [385] "Dim382" "Dim383" "Dim384" "Dim385" "Dim386" "Dim387" "Dim388" "Dim389"

161 ## [393] "Dim390" "Dim391" "Dim392" "Dim393" "Dim394" "Dim395" "Dim396" "Dim397"
162 ## [401] "Dim398" "Dim399" "Dim400" "Dim401" "Dim402" "Dim403" "Dim404" "Dim405"
163 ## [409] "Dim406" "Dim407" "Dim408" "Dim409" "Dim410" "Dim411" "Dim412" "Dim413"
164 ## [417] "Dim414" "Dim415" "Dim416" "Dim417" "Dim418" "Dim419" "Dim420" "Dim421"
165 ## [425] "Dim422" "Dim423" "Dim424" "Dim425" "Dim426" "Dim427" "Dim428" "Dim429"
166 ## [433] "Dim430" "Dim431" "Dim432" "Dim433" "Dim434" "Dim435" "Dim436" "Dim437"
167 ## [441] "Dim438" "Dim439" "Dim440" "Dim441" "Dim442" "Dim443" "Dim444" "Dim445"
168 ## [449] "Dim446" "Dim447" "Dim448" "Dim449" "Dim450" "Dim451" "Dim452" "Dim453"
169 ## [457] "Dim454" "Dim455" "Dim456" "Dim457" "Dim458" "Dim459" "Dim460" "Dim461"
170 ## [465] "Dim462" "Dim463" "Dim464" "Dim465" "Dim466" "Dim467" "Dim468" "Dim469"
171 ## [473] "Dim470" "Dim471" "Dim472" "Dim473" "Dim474" "Dim475" "Dim476" "Dim477"
172 ## [481] "Dim478" "Dim479" "Dim480" "Dim481" "Dim482" "Dim483" "Dim484" "Dim485"
173 ## [489] "Dim486" "Dim487" "Dim488" "Dim489" "Dim490" "Dim491" "Dim492" "Dim493"
174 ## [497] "Dim494" "Dim495" "Dim496" "Dim497" "Dim498" "Dim499" "Dim500" "Dim501"
175 ## [505] "Dim502" "Dim503" "Dim504" "Dim505" "Dim506" "Dim507" "Dim508" "Dim509"
176 ## [513] "Dim510" "Dim511" "Dim512" "Dim513" "Dim514" "Dim515" "Dim516" "Dim517"
177 ## [521] "Dim518" "Dim519" "Dim520" "Dim521" "Dim522" "Dim523" "Dim524" "Dim525"
178 ## [529] "Dim526" "Dim527" "Dim528" "Dim529" "Dim530" "Dim531" "Dim532" "Dim533"
179 ## [537] "Dim534" "Dim535" "Dim536" "Dim537" "Dim538" "Dim539" "Dim540" "Dim541"
180 ## [545] "Dim542" "Dim543" "Dim544" "Dim545" "Dim546" "Dim547" "Dim548" "Dim549"
181 ## [553] "Dim550" "Dim551" "Dim552" "Dim553" "Dim554" "Dim555" "Dim556" "Dim557"
182 ## [561] "Dim558" "Dim559" "Dim560" "Dim561" "Dim562" "Dim563" "Dim564" "Dim565"
183 ## [569] "Dim566" "Dim567" "Dim568" "Dim569" "Dim570" "Dim571" "Dim572" "Dim573"
184 ## [577] "Dim574" "Dim575" "Dim576" "Dim577" "Dim578" "Dim579" "Dim580" "Dim581"
185 ## [585] "Dim582" "Dim583" "Dim584" "Dim585" "Dim586" "Dim587" "Dim588" "Dim589"
186 ## [593] "Dim590" "Dim591" "Dim592" "Dim593" "Dim594" "Dim595" "Dim596" "Dim597"
187 ## [601] "Dim598" "Dim599" "Dim600" "Dim601" "Dim602" "Dim603" "Dim604" "Dim605"


```
188 ## [609] "Dim606" "Dim607" "Dim608" "Dim609" "Dim610" "Dim611" "Dim612" "Dim613"
189 ## [617] "Dim614" "Dim615" "Dim616" "Dim617" "Dim618" "Dim619" "Dim620" "Dim621"
190 ## [625] "Dim622" "Dim623" "Dim624" "Dim625" "Dim626" "Dim627" "Dim628" "Dim629"
191 ## [633] "Dim630" "Dim631" "Dim632" "Dim633" "Dim634" "Dim635" "Dim636" "Dim637"
192 ## [641] "Dim638" "Dim639" "Dim640" "Dim641" "Dim642" "Dim643" "Dim644" "Dim645"
193 ## [649] "Dim646" "Dim647" "Dim648" "Dim649" "Dim650" "Dim651" "Dim652" "Dim653"
194 ## [657] "Dim654" "Dim655" "Dim656" "Dim657" "Dim658" "Dim659" "Dim660" "Dim661"
195 ## [665] "Dim662" "Dim663" "Dim664" "Dim665" "Dim666" "Dim667" "Dim668" "Dim669"
196 ## [673] "Dim670" "Dim671" "Dim672" "Dim673" "Dim674" "Dim675" "Dim676" "Dim677"
197 ## [681] "Dim678" "Dim679" "Dim680" "Dim681" "Dim682" "Dim683" "Dim684" "Dim685"
198 ## [689] "Dim686" "Dim687" "Dim688" "Dim689" "Dim690" "Dim691" "Dim692" "Dim693"
199 ## [697] "Dim694" "Dim695" "Dim696" "Dim697" "Dim698" "Dim699" "Dim700" "Dim701"
200 ## [705] "Dim702" "Dim703" "Dim704" "Dim705" "Dim706" "Dim707" "Dim708" "Dim709"
201 ## [713] "Dim710" "Dim711" "Dim712" "Dim713" "Dim714" "Dim715" "Dim716" "Dim717"
202 ## [721] "Dim718" "Dim719" "Dim720" "Dim721" "Dim722" "Dim723" "Dim724" "Dim725"
203 ## [729] "Dim726" "Dim727" "Dim728" "Dim729" "Dim730" "Dim731" "Dim732" "Dim733"
204 ## [737] "Dim734" "Dim735" "Dim736" "Dim737" "Dim738" "Dim739" "Dim740" "Dim741"
205 ## [745] "Dim742" "Dim743" "Dim744" "Dim745" "Dim746" "Dim747" "Dim748" "Dim749"
206 ## [753] "Dim750" "Dim751" "Dim752" "Dim753" "Dim754" "Dim755" "Dim756" "Dim757"
207 ## [761] "Dim758" "Dim759" "Dim760" "Dim761" "Dim762" "Dim763" "Dim764" "Dim765"
208 ## [769] "Dim766" "Dim767" "Dim768"

209 ## tibble [1,000 x 769] (S3: tbl_df/tbl/data.frame)
210 ## $ Label : chr [1:1000] "hate" "hate" "nothate" "nothate" ...
211 ## $ Dim1 : num [1:1000] -0.0795 -0.0333 0.0155 0.0426 -0.011 ...
212 ## $ Dim2 : num [1:1000] 0.0283 0.0924 0.0863 0.1245 0.0967 ...
213 ## $ Dim3 : num [1:1000] 0.08579 0.00733 0.06838 -0.00661 0.05425 ...
214 ## $ Dim4 : num [1:1000] -0.2576 0.0412 -0.1447 -0.0421 -0.1275 ...
```

```
215 ## $ Dim5 : num [1:1000] 0.1991 0.4505 0.0322 0.3567 0.1109 ...
216 ## $ Dim6 : num [1:1000] -0.10062 -0.20682 -0.00324 -0.08741 0.24727 ...
217 ## $ Dim7 : num [1:1000] 0.0353 0.0507 0.0376 0.074 0.0456 ...
218 ## $ Dim8 : num [1:1000] 0.1016 0.0272 0.03 0.0806 -0.0162 ...
219 ## $ Dim9 : num [1:1000] 0.0938 -0.016 0.028 0.0128 0.0148 ...
220 ## $ Dim10 : num [1:1000] -0.04394 -0.05269 0.05363 -0.00703 0.05217 ...
221 ## $ Dim11 : num [1:1000] -0.0656 -0.0668 -0.0663 -0.1644 -0.1628 ...
222 ## $ Dim12 : num [1:1000] -0.0946 -0.0786 -0.0739 -0.1072 -0.167 ...
223 ## $ Dim13 : num [1:1000] 0.09203 0.01265 -0.0193 0.04477 -0.00249 ...
224 ## $ Dim14 : num [1:1000] -0.1079 -0.0969 -0.0741 0.1308 -0.0747 ...
225 ## $ Dim15 : num [1:1000] -0.0318 0.0475 0.0681 -0.1103 0.1283 ...
226 ## $ Dim16 : num [1:1000] 0.0843 0.1851 -0.1631 0.1157 -0.1573 ...
227 ## $ Dim17 : num [1:1000] 0.2017 -0.0898 0.1335 -0.0527 0.0325 ...
228 ## $ Dim18 : num [1:1000] 0.0362 -0.0144 0.0473 -0.0365 0.0795 ...
229 ## $ Dim19 : num [1:1000] -0.022 -0.0509 0.1179 0.0929 -0.0228 ...
230 ## $ Dim20 : num [1:1000] 0.165 -0.032 0.158 0.152 -0.026 ...
231 ## $ Dim21 : num [1:1000] -0.1072 -0.0415 -0.0726 -0.0565 -0.0726 ...
232 ## $ Dim22 : num [1:1000] 0.09555 0.06618 0.14415 -0.00492 0.22261 ...
233 ## $ Dim23 : num [1:1000] -0.1781 -0.0181 -0.0236 -0.0777 -0.0872 ...
234 ## $ Dim24 : num [1:1000] 0.14796 0.12122 0.05944 0.00402 0.04334 ...
235 ## $ Dim25 : num [1:1000] -0.0202 -0.0354 -0.0415 -0.0504 -0.1469 ...
236 ## $ Dim26 : num [1:1000] 0.00976 0.04094 0.08391 -0.01513 0.12582 ...
237 ## $ Dim27 : num [1:1000] 0.03 0.0165 0.151 0.1833 0.0762 ...
238 ## $ Dim28 : num [1:1000] 0.0132 -0.1301 0.0937 -0.0317 -0.0918 ...
239 ## $ Dim29 : num [1:1000] -0.03667 0.02588 0.00581 0.08772 -0.0529 ...
240 ## $ Dim30 : num [1:1000] 0.0693 0.1139 -0.0113 -0.0376 -0.1253 ...
241 ## $ Dim31 : num [1:1000] -0.13636 0.00998 -0.0944 -0.11835 -0.0611 ...
```

```
242 ## $ Dim32 : num [1:1000] -0.1024 -0.0828 -0.0233 0.0132 -0.0229 ...
243 ## $ Dim33 : num [1:1000] 0.055908 0.000795 0.040355 -0.006121 0.07767 ...
244 ## $ Dim34 : num [1:1000] -0.01342 -0.03566 0.08336 -0.00649 0.09765 ...
245 ## $ Dim35 : num [1:1000] 0.01331 0.00535 -0.04677 -0.05681 -0.07868 ...
246 ## $ Dim36 : num [1:1000] 0.01555 0.0425 -0.0151 -0.08599 -0.00778 ...
247 ## $ Dim37 : num [1:1000] 0.0995 0.139 0.0783 0.0373 0.1383 ...
248 ## $ Dim38 : num [1:1000] 0.04237 0.00826 0.00658 0.01017 -0.00746 ...
249 ## $ Dim39 : num [1:1000] 0.345 0.1914 0.2469 0.0745 0.1925 ...
250 ## $ Dim40 : num [1:1000] 0.1028 0.0046 0.0649 -0.0728 0.0183 ...
251 ## $ Dim41 : num [1:1000] -0.1233 -0.1269 0.0897 0.0106 -0.0169 ...
252 ## $ Dim42 : num [1:1000] -0.2174 -0.0419 -0.1553 -0.2404 0.0491 ...
253 ## $ Dim43 : num [1:1000] -0.00121 0.05386 -0.0154 0.00676 -0.07798 ...
254 ## $ Dim44 : num [1:1000] -0.0293 -0.0378 0.033 -0.0028 0.0292 ...
255 ## $ Dim45 : num [1:1000] 0.1032 0.0318 0.0107 0.0263 0.0378 ...
256 ## $ Dim46 : num [1:1000] 0.0326 -0.0295 0.0209 0.0135 0.0691 ...
257 ## $ Dim47 : num [1:1000] -0.1591 -0.0431 -0.1508 -0.1071 -0.1156 ...
258 ## $ Dim48 : num [1:1000] 0.1301 -0.0315 -0.1125 0.0825 -0.0899 ...
259 ## $ Dim49 : num [1:1000] 0.04845 -0.06322 -0.00179 0.06586 0.048 ...
260 ## $ Dim50 : num [1:1000] 0.0839 -0.0281 -0.0133 0.0114 -0.1345 ...
261 ## $ Dim51 : num [1:1000] -0.0917 0.0868 -0.0278 -0.0355 -0.0683 ...
262 ## $ Dim52 : num [1:1000] -0.0234 -0.0176 0.1633 0.1193 0.0173 ...
263 ## $ Dim53 : num [1:1000] -0.022 0.0954 -0.012 0.0484 0.1322 ...
264 ## $ Dim54 : num [1:1000] -1.51e-02 -7.33e-02 1.75e-05 3.63e-03 7.05e-02 ...
265 ## $ Dim55 : num [1:1000] 0.0408 0.0366 -0.0173 -0.0549 0.0221 ...
266 ## $ Dim56 : num [1:1000] 0.0137 0.0543 0.046 -0.0196 0.0293 ...
267 ## $ Dim57 : num [1:1000] 0.072 0.019 0.1006 0.0708 0.1566 ...
268 ## $ Dim58 : num [1:1000] 0.233 0.193 0.162 0.373 0.231 ...
```

```
269 ## $ Dim59 : num [1:1000] 0.0733 -0.055 0.0294 0.0662 0.0432 ...
270 ## $ Dim60 : num [1:1000] 0.01502 -0.09608 -0.01706 0.00262 -0.01736 ...
271 ## $ Dim61 : num [1:1000] 0.01 0.0385 -0.0233 -0.0503 0.052 ...
272 ## $ Dim62 : num [1:1000] 0.4931 0.4505 0.0755 0.4037 -0.1958 ...
273 ## $ Dim63 : num [1:1000] -0.1841 -0.0605 -0.0428 -0.0792 -0.1128 ...
274 ## $ Dim64 : num [1:1000] 0.0996 -0.0166 0.0474 -0.0503 0.1249 ...
275 ## $ Dim65 : num [1:1000] -0.0798 -0.00528 0.02712 0.03789 0.08076 ...
276 ## $ Dim66 : num [1:1000] 0.0348 0.016 -0.0105 -0.0118 -0.0121 ...
277 ## $ Dim67 : num [1:1000] 0.09249 -0.02617 0.01732 0.00648 0.09461 ...
278 ## $ Dim68 : num [1:1000] -0.0602 -0.2076 -0.0604 -0.0934 0.0936 ...
279 ## $ Dim69 : num [1:1000] 0.0315 0.0986 -0.0499 0.0323 0.0737 ...
280 ## $ Dim70 : num [1:1000] -0.0503 0.0767 0.0641 0.082 0.1444 ...
281 ## $ Dim71 : num [1:1000] 0.0941 0.0891 -0.0412 0.0391 -0.053 ...
282 ## $ Dim72 : num [1:1000] -0.1152 0.0205 -0.0256 -0.0618 -0.0503 ...
283 ## $ Dim73 : num [1:1000] -0.0218 0.0898 0.0409 -0.0411 0.0109 ...
284 ## $ Dim74 : num [1:1000] -0.047 -0.0574 0.0274 -0.1004 0.0533 ...
285 ## $ Dim75 : num [1:1000] 0.05422 -0.00118 0.06607 0.01979 0.13762 ...
286 ## $ Dim76 : num [1:1000] 0.1064 -0.1159 0.0546 0.0536 0.0835 ...
287 ## $ Dim77 : num [1:1000] 0.0187 0.065 0.0521 0.0837 0.0691 ...
288 ## $ Dim78 : num [1:1000] -4.89 -2.42 -5.9 -3.63 -5.69 ...
289 ## $ Dim79 : num [1:1000] -0.0492 -0.0797 -0.1108 0.0334 -0.0179 ...
290 ## $ Dim80 : num [1:1000] 0.1146 0.00874 -0.09434 0.02867 0.07683 ...
291 ## $ Dim81 : num [1:1000] -0.00173 0.0534 0.10269 0.11348 0.08096 ...
292 ## $ Dim82 : num [1:1000] -0.2051 -0.0792 -0.0913 -0.0359 -0.0849 ...
293 ## $ Dim83 : num [1:1000] 0.2152 0.0969 0.6394 0.4623 0.5441 ...
294 ## $ Dim84 : num [1:1000] -0.000328 0.034866 -0.061725 0.057915 -0.033154 ...
295 ## $ Dim85 : num [1:1000] 0.0418 0.1306 0.0755 0.0558 -0.0413 ...
```

```

296 ## $ Dim86 : num [1:1000] -0.1317 0.3587 -0.0523 0.028 -0.0539 ...
297 ## $ Dim87 : num [1:1000] 0.0286 0.09665 -0.00575 -0.02233 0.0037 ...
298 ## $ Dim88 : num [1:1000] -0.0774 0.173 -0.1015 -0.0337 0.0413 ...
299 ## $ Dim89 : num [1:1000] -0.01247 0.0665 -0.00921 0.0184 0.01273 ...
300 ## $ Dim90 : num [1:1000] 0.03029 -0.09836 0.00135 -0.01018 0.13343 ...
301 ## $ Dim91 : num [1:1000] -0.0164 0.0291 0.1429 0.1461 0.0221 ...
302 ## $ Dim92 : num [1:1000] 0.0293 -0.0356 -0.0273 0.0174 0.1431 ...
303 ## $ Dim93 : num [1:1000] -0.01328 -0.00859 -0.03453 -0.01491 0.01285 ...
304 ## $ Dim94 : num [1:1000] 0.06097 0.00899 0.05293 0.03094 0.21327 ...
305 ## $ Dim95 : num [1:1000] 0.0444 0.04561 0.06597 0.08142 -0.00476 ...
306 ## $ Dim96 : num [1:1000] -0.00381 0.0402 0.09011 -0.01924 -0.04217 ...
307 ## $ Dim97 : num [1:1000] -0.036561 0.027386 -0.020688 -0.000983 -0.028573 ...
308 ## $ Dim98 : num [1:1000] 0.5007 -0.061 0.5236 0.0664 0.1875 ...
309 ## [list output truncated]

310 ## Recipe
311 ##
312 ## Inputs:
313 ##
314 ##      role #variables
315 ## outcome      1
316 ## predictor    768
317 ##
318 ## Operations:
319 ##
320 ## Centering and scaling for all_of(numeric)

321 ## Recipe

```

```

322 ##
323 ## Inputs:
324 ##
325 ##      role #variables
326 ##      outcome      1
327 ##      predictor      768
328 ##
329 ## Training data contained 1000 data points and no missing data.
330 ##
331 ## Operations:
332 ##
333 ## Centering and scaling for Dim1, Dim2, Dim3, Dim4, Dim5, Dim6, Dim7, Dim8,... [trained
334 ## # A tibble: 1,000 x 769
335 ##   Label      Dim1  Dim2  Dim3  Dim4  Dim5  Dim6  Dim7  Dim8  Dim9
336 ##   <fct>    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
337 ## 1 hate    -1.90  -0.358  1.45  -2.00  0.880 -0.750 -0.0661  1.34  0.601
338 ## 2 hate    -0.871  0.695 -0.486  1.83  2.46  -1.84  0.312  0.106 -1.12
339 ## 3 nothate  0.219  0.594  1.02  -0.554 -0.170  0.254 -0.00966  0.151 -0.432
340 ## 4 nothate  0.823  1.22  -0.830  0.762  1.87  -0.613  0.887  0.992 -0.670
341 ## 5 nothate -0.374  0.765  0.675  -0.334  0.326  2.84  0.187  -0.616 -0.639
342 ## 6 nothate  1.07  0.423 -0.0821 -0.0748 -0.729 -0.641  2.52  -0.149 -0.810
343 ## 7 nothate  0.215  0.894 -1.88  -0.962  2.49  -0.726  2.51  -0.405 -0.442
344 ## 8 hate    0.00453 -0.214 -0.158  -0.406  0.845  0.421 -1.34  0.624 -1.53
345 ## 9 hate    -0.773  0.261  1.12  0.644  -0.0518  0.752 -0.427  -0.577 -0.517
346 ## 10 nothate -1.41  0.251  0.887  -0.459  1.07  -1.48  1.64  -0.127  1.38
347 ## # ... with 990 more rows, and 759 more variables: Dim10 <dbl>, Dim11 <dbl>,
348 ## #   Dim12 <dbl>, Dim13 <dbl>, Dim14 <dbl>, Dim15 <dbl>, Dim16 <dbl>,

```

```
349 ## # Dim17 <dbl>, Dim18 <dbl>, Dim19 <dbl>, Dim20 <dbl>, Dim21 <dbl>,  
350 ## # Dim22 <dbl>, Dim23 <dbl>, Dim24 <dbl>, Dim25 <dbl>, Dim26 <dbl>,  
351 ## # Dim27 <dbl>, Dim28 <dbl>, Dim29 <dbl>, Dim30 <dbl>, Dim31 <dbl>,  
352 ## # Dim32 <dbl>, Dim33 <dbl>, Dim34 <dbl>, Dim35 <dbl>, Dim36 <dbl>,  
353 ## # Dim37 <dbl>, Dim38 <dbl>, Dim39 <dbl>, Dim40 <dbl>, Dim41 <dbl>, ...
```

354 Logistic regression

```
355 ## Warning in train.recipe(blueprint_12, data = data_12_tr, method = "glm", : The  
356 ## metric "Accuracy" was not in the result set. logLoss will be used instead.
```

```
357 ## Warning: glm.fit: algorithm did not converge
```

```
358 ## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :  
359 ## prediction from a rank-deficient fit may be misleading
```

```
360
```

```
361 ## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :  
362 ## prediction from a rank-deficient fit may be misleading
```

```
363 ## Warning: glm.fit: algorithm did not converge
```

```
364 ## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :  
365 ## prediction from a rank-deficient fit may be misleading
```

```
366
```

```
367 ## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :  
368 ## prediction from a rank-deficient fit may be misleading
```

```
369 ## Warning: glm.fit: algorithm did not converge
```

```
370 ## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
371 ## prediction from a rank-deficient fit may be misleading
372
373 ## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
374 ## prediction from a rank-deficient fit may be misleading
375
376 ## Warning: glm.fit: algorithm did not converge
377
378
379 ## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
380 ## prediction from a rank-deficient fit may be misleading
381
382 ## Warning: glm.fit: algorithm did not converge
383
384
385 ## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
386 ## prediction from a rank-deficient fit may be misleading
387
388 ## Warning: glm.fit: algorithm did not converge
389
390
391 ## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
392 ## prediction from a rank-deficient fit may be misleading
```



```
393 ## Warning: glm.fit: algorithm did not converge

394 ## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
395 ## prediction from a rank-deficient fit may be misleading
396
397 ## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
398 ## prediction from a rank-deficient fit may be misleading

399 ## Warning: glm.fit: algorithm did not converge

400 ## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
401 ## prediction from a rank-deficient fit may be misleading
402
403 ## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
404 ## prediction from a rank-deficient fit may be misleading

405 ## Warning: glm.fit: algorithm did not converge

406 ## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
407 ## prediction from a rank-deficient fit may be misleading
408
409 ## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
410 ## prediction from a rank-deficient fit may be misleading

411 ## Warning: glm.fit: algorithm did not converge

412 ## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
413 ## prediction from a rank-deficient fit may be misleading
414
415 ## Warning in predict.lm(object, newdata, se.fit, scale = 1, type = if (type == :
416 ## prediction from a rank-deficient fit may be misleading
```

```
417 ## Warning: glm.fit: algorithm did not converge

418 ## Warning: glm.fit: fitted probabilities numerically 0 or 1 occurred

419 ## Generalized Linear Model
420 ##
421 ## 800 samples
422 ## 768 predictors
423 ## 2 classes: 'hate', 'nothate'
424 ##
425 ## Recipe steps: normalize
426 ## Resampling: Cross-Validated (10 fold)
427 ## Summary of sample sizes: 720, 720, 720, 720, 720, 720, ...
428 ## Resampling results:
429 ##
430 ## logLoss
431 ## 17

432 ## hate nothate
433 ## 1 1.5e-10 1.0e+00
434 ## 2 1.7e-07 1.0e+00
435 ## 3 2.2e-16 1.0e+00
436 ## 4 2.2e-16 1.0e+00
437 ## 5 2.2e-16 1.0e+00
438 ## 6 2.2e-16 1.0e+00
439 ## 7 2.2e-16 1.0e+00
440 ## 8 1.0e+00 2.2e-16
441 ## 9 2.2e-16 1.0e+00
```

442 ## 10 1.0e+00 1.1e-03
443 ## 11 2.2e-16 1.0e+00
444 ## 12 1.0e+00 2.2e-16
445 ## 13 1.0e+00 2.2e-16
446 ## 14 1.0e+00 2.2e-16
447 ## 15 1.0e+00 2.2e-16
448 ## 16 1.0e+00 2.2e-16
449 ## 17 3.1e-06 1.0e+00
450 ## 18 1.0e+00 2.2e-16
451 ## 19 2.2e-16 1.0e+00
452 ## 20 1.0e+00 2.2e-16
453 ## 21 1.0e+00 2.2e-16
454 ## 22 1.0e+00 2.2e-16
455 ## 23 2.2e-16 1.0e+00
456 ## 24 2.2e-16 1.0e+00
457 ## 25 2.2e-16 1.0e+00
458 ## 26 2.2e-16 1.0e+00
459 ## 27 1.0e+00 2.2e-16
460 ## 28 2.2e-16 1.0e+00
461 ## 29 3.9e-04 1.0e+00
462 ## 30 2.2e-16 1.0e+00
463 ## 31 2.2e-16 1.0e+00
464 ## 32 1.0e+00 2.2e-16
465 ## 33 5.8e-05 1.0e+00
466 ## 34 2.2e-16 1.0e+00
467 ## 35 1.0e+00 2.2e-16
468 ## 36 1.0e+00 2.2e-16

469 ## 37 1.0e+00 2.2e-16
470 ## 38 1.0e+00 2.2e-16
471 ## 39 2.2e-16 1.0e+00
472 ## 40 1.0e+00 2.2e-16
473 ## 41 2.2e-16 1.0e+00
474 ## 42 6.4e-01 3.6e-01
475 ## 43 1.0e+00 2.2e-16
476 ## 44 1.0e+00 2.2e-16
477 ## 45 1.0e+00 2.2e-16
478 ## 46 2.2e-16 1.0e+00
479 ## 47 1.0e+00 2.2e-16
480 ## 48 1.0e+00 2.2e-16
481 ## 49 1.0e+00 2.2e-16
482 ## 50 2.2e-16 1.0e+00
483 ## 51 1.0e+00 2.2e-16
484 ## 52 1.0e+00 2.2e-16
485 ## 53 2.2e-16 1.0e+00
486 ## 54 2.2e-16 1.0e+00
487 ## 55 1.0e+00 2.2e-16
488 ## 56 1.0e+00 2.2e-16
489 ## 57 1.0e+00 2.2e-16
490 ## 58 3.3e-01 6.7e-01
491 ## 59 1.0e+00 2.2e-16
492 ## 60 1.0e+00 1.5e-12
493 ## 61 1.0e+00 2.2e-16
494 ## 62 2.2e-16 1.0e+00
495 ## 63 2.2e-16 1.0e+00

496	##	64	2.2e-16	1.0e+00
497	##	65	1.0e+00	3.1e-06
498	##	66	1.0e+00	2.2e-16
499	##	67	1.6e-11	1.0e+00
500	##	68	3.3e-04	1.0e+00
501	##	69	1.0e+00	2.2e-16
502	##	70	1.0e+00	2.2e-16
503	##	71	2.2e-16	1.0e+00
504	##	72	1.2e-06	1.0e+00
505	##	73	2.2e-16	1.0e+00
506	##	74	2.2e-16	1.0e+00
507	##	75	1.0e+00	2.2e-16
508	##	76	1.7e-06	1.0e+00
509	##	77	1.0e+00	2.2e-16
510	##	78	1.0e+00	2.2e-16
511	##	79	1.0e+00	2.2e-16
512	##	80	1.0e+00	2.2e-16
513	##	81	1.0e+00	2.3e-08
514	##	82	1.0e+00	2.2e-16
515	##	83	2.2e-16	1.0e+00
516	##	84	1.0e+00	2.2e-16
517	##	85	1.0e-06	1.0e+00
518	##	86	1.0e+00	2.2e-16
519	##	87	1.1e-07	1.0e+00
520	##	88	2.2e-16	1.0e+00
521	##	89	2.2e-16	1.0e+00
522	##	90	1.0e+00	2.0e-04

523 ## 91 2.2e-16 1.0e+00
524 ## 92 6.4e-08 1.0e+00
525 ## 93 1.0e+00 2.2e-16
526 ## 94 1.0e+00 2.2e-16
527 ## 95 1.0e+00 2.2e-16
528 ## 96 4.9e-02 9.5e-01
529 ## 97 1.0e+00 2.2e-16
530 ## 98 1.0e+00 2.2e-16
531 ## 99 1.0e+00 2.2e-16
532 ## 100 1.0e+00 1.3e-07
533 ## 101 2.2e-16 1.0e+00
534 ## 102 2.2e-16 1.0e+00
535 ## 103 2.2e-16 1.0e+00
536 ## 104 1.0e+00 2.2e-16
537 ## 105 2.2e-16 1.0e+00
538 ## 106 1.0e+00 2.2e-16
539 ## 107 1.0e+00 2.2e-16
540 ## 108 4.6e-11 1.0e+00
541 ## 109 1.0e+00 2.2e-16
542 ## 110 1.0e+00 2.2e-16
543 ## 111 1.0e+00 2.2e-16
544 ## 112 1.0e+00 2.2e-16
545 ## 113 1.0e+00 2.2e-16
546 ## 114 1.0e+00 2.2e-16
547 ## 115 2.2e-16 1.0e+00
548 ## 116 2.2e-16 1.0e+00
549 ## 117 9.5e-02 9.0e-01

```
550 ## 118 2.2e-16 1.0e+00
551 ## 119 1.0e+00 2.2e-16
552 ## 120 2.2e-16 1.0e+00
553 ## 121 1.0e+00 2.2e-16
554 ## 122 2.2e-16 1.0e+00
555 ## 123 2.2e-16 1.0e+00
556 ## 124 2.2e-16 1.0e+00
557 ## 125 1.0e+00 2.2e-16
558 ## 126 1.1e-08 1.0e+00
559 ## 127 6.1e-13 1.0e+00
560 ## 128 1.0e+00 2.2e-16
561 ## 129 2.2e-16 1.0e+00
562 ## 130 2.2e-16 1.0e+00
563 ## 131 2.2e-16 1.0e+00
564 ## 132 2.2e-16 1.0e+00
565 ## 133 2.2e-16 1.0e+00
566 ## 134 1.0e+00 2.2e-16
567 ## 135 1.0e+00 2.2e-16
568 ## 136 2.2e-10 1.0e+00
569 ## 137 1.0e+00 4.9e-07
570 ## 138 2.2e-16 1.0e+00
571 ## 139 2.2e-16 1.0e+00
572 ## 140 2.2e-16 1.0e+00
573 ## 141 2.2e-16 1.0e+00
574 ## 142 2.2e-16 1.0e+00
575 ## 143 1.0e+00 2.2e-16
576 ## 144 2.2e-16 1.0e+00
```

577 ## 145 4.0e-07 1.0e+00
578 ## 146 1.0e+00 2.2e-16
579 ## 147 1.0e+00 2.2e-16
580 ## 148 1.0e+00 2.2e-16
581 ## 149 1.0e+00 2.2e-16
582 ## 150 2.2e-16 1.0e+00
583 ## 151 1.0e+00 2.2e-16
584 ## 152 2.2e-16 1.0e+00
585 ## 153 3.3e-09 1.0e+00
586 ## 154 1.0e+00 2.2e-16
587 ## 155 1.0e+00 2.2e-16
588 ## 156 1.0e+00 2.2e-16
589 ## 157 2.2e-16 1.0e+00
590 ## 158 2.2e-16 1.0e+00
591 ## 159 2.2e-16 1.0e+00
592 ## 160 1.0e+00 2.2e-03
593 ## 161 1.0e+00 2.2e-16
594 ## 162 1.6e-03 1.0e+00
595 ## 163 1.0e+00 2.3e-11
596 ## 164 1.0e+00 2.2e-16
597 ## 165 1.0e+00 2.2e-16
598 ## 166 1.0e+00 2.2e-16
599 ## 167 3.9e-08 1.0e+00
600 ## 168 2.2e-16 1.0e+00
601 ## 169 1.0e+00 1.2e-06
602 ## 170 1.0e+00 2.2e-16
603 ## 171 1.0e+00 2.2e-16


```
604 ## 172 1.0e+00 2.2e-16
605 ## 173 1.0e+00 2.2e-16
606 ## 174 1.0e+00 2.2e-16
607 ## 175 1.0e+00 2.2e-16
608 ## 176 1.0e+00 2.2e-16
609 ## 177 2.2e-16 1.0e+00
610 ## 178 1.0e+00 2.2e-16
611 ## 179 1.0e+00 2.2e-16
612 ## 180 2.2e-16 1.0e+00
613 ## 181 2.2e-16 1.0e+00
614 ## 182 1.0e+00 2.2e-16
615 ## 183 1.0e+00 2.2e-16
616 ## 184 1.0e+00 2.2e-16
617 ## 185 1.0e+00 2.2e-16
618 ## 186 1.0e+00 2.2e-16
619 ## 187 1.0e+00 2.2e-16
620 ## 188 1.0e+00 2.2e-16
621 ## 189 2.2e-16 1.0e+00
622 ## 190 1.0e+00 5.5e-10
623 ## 191 1.0e+00 2.2e-16
624 ## 192 2.2e-16 1.0e+00
625 ## 193 2.2e-16 1.0e+00
626 ## 194 1.0e+00 2.2e-16
627 ## 195 1.0e+00 2.2e-16
628 ## 196 1.0e+00 2.2e-16
629 ## 197 1.0e+00 2.2e-16
630 ## 198 2.2e-16 1.0e+00
```

631 ## 199 3.6e-05 1.0e+00

632 ## 200 1.0e+00 2.2e-16

633 ## pred_class_12

634 ## 0 1

635 ## hate 49 65

636 ## nothate 42 44

637 Ridge regression

638 ## alpha lambda

639 ## 1 0 0e+00

640 ## 2 0 1e-05

641 ## 3 0 2e-05

642 ## 4 0 3e-05

643 ## 5 0 4e-05

644 ## 6 0 5e-05

645 ## 7 0 6e-05

646 ## 8 0 7e-05

647 ## 9 0 8e-05

648 ## 10 0 9e-05

649 ## 11 0 1e-04

650 ## 12 0 5e-03

651 ## 13 0 1e-02

652 ## 14 0 5e-02

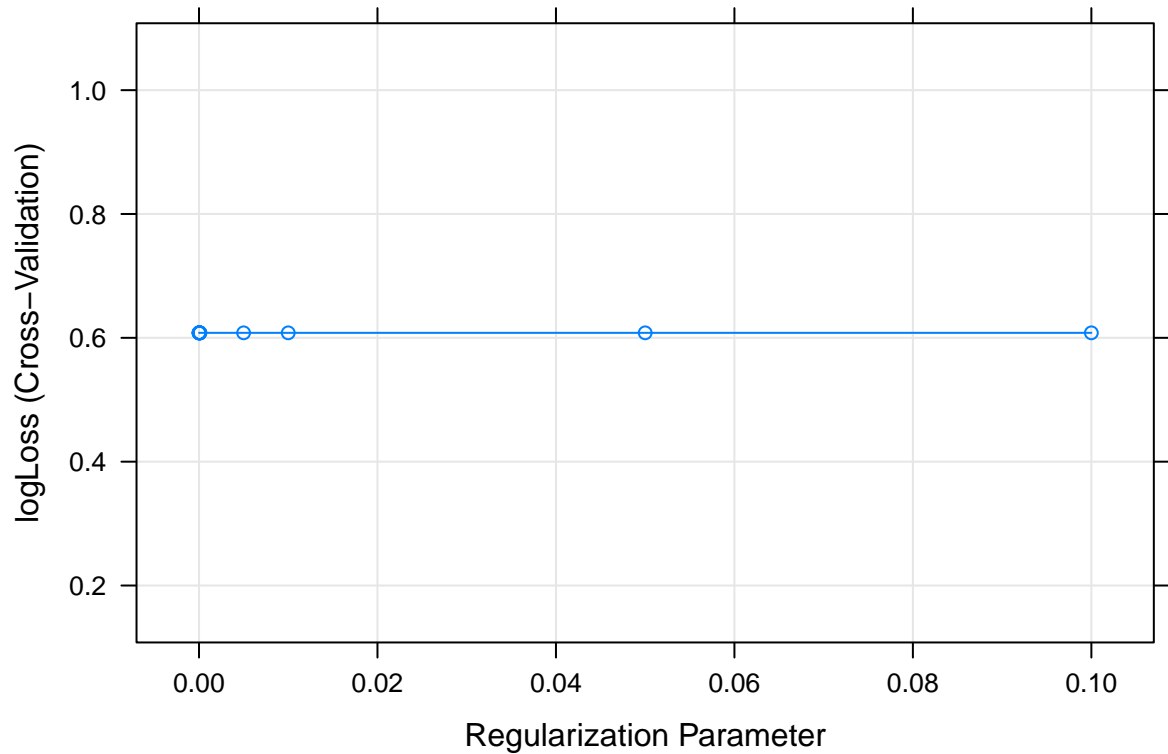
653 ## 15 0 1e-01

654 ## glmnet

655 ##

```
656 ## 800 samples
657 ## 768 predictors
658 ## 2 classes: 'hate', 'nothate'
659 ##
660 ## Recipe steps: normalize
661 ## Resampling: Cross-Validated (10 fold)
662 ## Summary of sample sizes: 720, 720, 720, 720, 720, 720, ...
663 ## Resampling results across tuning parameters:
664 ##
665 ##   lambda  logLoss
666 ##   0e+00   0.61
667 ##   1e-05   0.61
668 ##   2e-05   0.61
669 ##   3e-05   0.61
670 ##   4e-05   0.61
671 ##   5e-05   0.61
672 ##   6e-05   0.61
673 ##   7e-05   0.61
674 ##   8e-05   0.61
675 ##   9e-05   0.61
676 ##   1e-04   0.61
677 ##   5e-03   0.61
678 ##   1e-02   0.61
679 ##   5e-02   0.61
680 ##   1e-01   0.61
681 ##
682 ## Tuning parameter 'alpha' was held constant at a value of 0
```

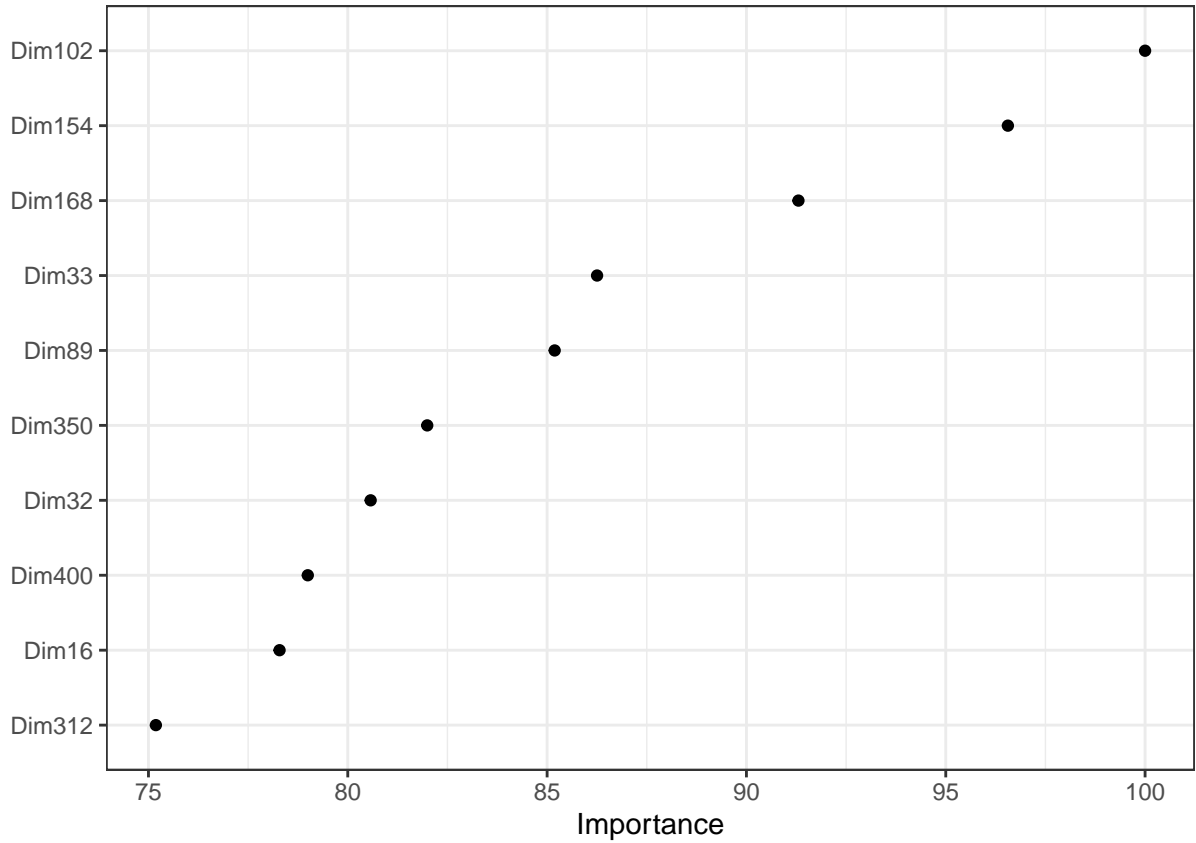
```
683 ## logLoss was used to select the optimal model using the smallest value.  
684 ## The final values used for the model were alpha = 0 and lambda = 0.1.
```



```
685
```

```
686 ##      alpha lambda  
687 ## 15      0      0.1
```

```
688 ##          pred_class_12  
689 ##              0  1  
690 ##    hate      32 82  
691 ##    nothate  50 36
```



692

693 ## [1] 769

694 ## [,1]

695 ## (Intercept) -0.33

696 ## Dim102 -0.15

697 ## Dim154 0.15

698 ## Dim168 0.14

699 ## Dim33 -0.13

700 ## Dim89 -0.13

701 ## Dim350 0.13

702 ## Dim32 -0.12

703 ## Dim400 -0.12

704 ## Dim16 0.12

705 **Logistic Regression with Lasso penalty**

706	##	alpha	lambda
707	## 1	1	0.016
708	## 2	1	0.016
709	## 3	1	0.016
710	## 4	1	0.016
711	## 5	1	0.016
712	## 6	1	0.016
713	## 7	1	0.016
714	## 8	1	0.016
715	## 9	1	0.016
716	## 10	1	0.016
717	## 11	1	0.016
718	## 12	1	0.016
719	## 13	1	0.016
720	## 14	1	0.016
721	## 15	1	0.016
722	## 16	1	0.016
723	## 17	1	0.016
724	## 18	1	0.016
725	## 19	1	0.016
726	## 20	1	0.016
727	## 21	1	0.016
728	## 22	1	0.016
729	## 23	1	0.016
730	## 24	1	0.016
731	## 25	1	0.016

732	## 26	1	0.016
733	## 27	1	0.016
734	## 28	1	0.016
735	## 29	1	0.016
736	## 30	1	0.016
737	## 31	1	0.016
738	## 32	1	0.016
739	## 33	1	0.016
740	## 34	1	0.016
741	## 35	1	0.016
742	## 36	1	0.016
743	## 37	1	0.016
744	## 38	1	0.016
745	## 39	1	0.016
746	## 40	1	0.016
747	## 41	1	0.016
748	## 42	1	0.016
749	## 43	1	0.016
750	## 44	1	0.016
751	## 45	1	0.016
752	## 46	1	0.016
753	## 47	1	0.016
754	## 48	1	0.016
755	## 49	1	0.016
756	## 50	1	0.016
757	## 51	1	0.017
758	## 52	1	0.017

759	## 53	1	0.017
760	## 54	1	0.017
761	## 55	1	0.017
762	## 56	1	0.017
763	## 57	1	0.017
764	## 58	1	0.017
765	## 59	1	0.017
766	## 60	1	0.017
767	## 61	1	0.017
768	## 62	1	0.017
769	## 63	1	0.017
770	## 64	1	0.017
771	## 65	1	0.017
772	## 66	1	0.017
773	## 67	1	0.017
774	## 68	1	0.017
775	## 69	1	0.017
776	## 70	1	0.017
777	## 71	1	0.017
778	## 72	1	0.017
779	## 73	1	0.017
780	## 74	1	0.017
781	## 75	1	0.017
782	## 76	1	0.017
783	## 77	1	0.017
784	## 78	1	0.017
785	## 79	1	0.017

786	## 80	1	0.017
787	## 81	1	0.017
788	## 82	1	0.017
789	## 83	1	0.017
790	## 84	1	0.017
791	## 85	1	0.017
792	## 86	1	0.017
793	## 87	1	0.017
794	## 88	1	0.017
795	## 89	1	0.017
796	## 90	1	0.017
797	## 91	1	0.017
798	## 92	1	0.017
799	## 93	1	0.017
800	## 94	1	0.017
801	## 95	1	0.017
802	## 96	1	0.017
803	## 97	1	0.017
804	## 98	1	0.017
805	## 99	1	0.017
806	## 100	1	0.017
807	## 101	1	0.017
808	## 102	1	0.017
809	## 103	1	0.017
810	## 104	1	0.017
811	## 105	1	0.017
812	## 106	1	0.017

813	## 107	1	0.017
814	## 108	1	0.017
815	## 109	1	0.017
816	## 110	1	0.017
817	## 111	1	0.017
818	## 112	1	0.017
819	## 113	1	0.017
820	## 114	1	0.017
821	## 115	1	0.017
822	## 116	1	0.017
823	## 117	1	0.017
824	## 118	1	0.017
825	## 119	1	0.017
826	## 120	1	0.017
827	## 121	1	0.017
828	## 122	1	0.017
829	## 123	1	0.017
830	## 124	1	0.017
831	## 125	1	0.017
832	## 126	1	0.017
833	## 127	1	0.017
834	## 128	1	0.017
835	## 129	1	0.017
836	## 130	1	0.017
837	## 131	1	0.017
838	## 132	1	0.017
839	## 133	1	0.017

840	## 134	1	0.017
841	## 135	1	0.017
842	## 136	1	0.017
843	## 137	1	0.017
844	## 138	1	0.017
845	## 139	1	0.017
846	## 140	1	0.017
847	## 141	1	0.017
848	## 142	1	0.017
849	## 143	1	0.017
850	## 144	1	0.017
851	## 145	1	0.017
852	## 146	1	0.017
853	## 147	1	0.017
854	## 148	1	0.017
855	## 149	1	0.017
856	## 150	1	0.017
857	## 151	1	0.018
858	## 152	1	0.018
859	## 153	1	0.018
860	## 154	1	0.018
861	## 155	1	0.018
862	## 156	1	0.018
863	## 157	1	0.018
864	## 158	1	0.018
865	## 159	1	0.018
866	## 160	1	0.018

867	## 161	1	0.018
868	## 162	1	0.018
869	## 163	1	0.018
870	## 164	1	0.018
871	## 165	1	0.018
872	## 166	1	0.018
873	## 167	1	0.018
874	## 168	1	0.018
875	## 169	1	0.018
876	## 170	1	0.018
877	## 171	1	0.018
878	## 172	1	0.018
879	## 173	1	0.018
880	## 174	1	0.018
881	## 175	1	0.018
882	## 176	1	0.018
883	## 177	1	0.018
884	## 178	1	0.018
885	## 179	1	0.018
886	## 180	1	0.018
887	## 181	1	0.018
888	## 182	1	0.018
889	## 183	1	0.018
890	## 184	1	0.018
891	## 185	1	0.018
892	## 186	1	0.018
893	## 187	1	0.018

```
894 ## 188      1  0.018
895 ## 189      1  0.018
896 ## 190      1  0.018
897 ## 191      1  0.018
898 ## 192      1  0.018
899 ## 193      1  0.018
900 ## 194      1  0.018
901 ## 195      1  0.018
902 ## 196      1  0.018
903 ## 197      1  0.018
904 ## 198      1  0.018
905 ## 199      1  0.018
906 ## 200      1  0.018
907 ## 201      1  0.018

908 ## glmnet
909 ##
910 ## 800 samples
911 ## 768 predictors
912 ##   2 classes: 'hate', 'nothate'
913 ##
914 ## Recipe steps: normalize
915 ## Resampling: Cross-Validated (10 fold)
916 ## Summary of sample sizes: 720, 720, 720, 720, 720, 720, ...
917 ## Resampling results across tuning parameters:
918 ##
919 ##   lambda  logLoss
920 ##   0.016   0.61
```

921	##	0.016	0.61
922	##	0.016	0.61
923	##	0.016	0.61
924	##	0.016	0.61
925	##	0.016	0.61
926	##	0.016	0.61
927	##	0.016	0.61
928	##	0.016	0.61
929	##	0.016	0.61
930	##	0.016	0.61
931	##	0.016	0.61
932	##	0.016	0.61
933	##	0.016	0.61
934	##	0.016	0.61
935	##	0.016	0.61
936	##	0.016	0.61
937	##	0.016	0.61
938	##	0.016	0.61
939	##	0.016	0.61
940	##	0.016	0.61
941	##	0.016	0.61
942	##	0.016	0.61
943	##	0.016	0.61
944	##	0.016	0.61
945	##	0.016	0.61
946	##	0.016	0.61
947	##	0.016	0.61

948	##	0.016	0.61
949	##	0.016	0.61
950	##	0.016	0.61
951	##	0.016	0.61
952	##	0.016	0.61
953	##	0.016	0.61
954	##	0.016	0.61
955	##	0.016	0.61
956	##	0.016	0.61
957	##	0.016	0.61
958	##	0.016	0.61
959	##	0.016	0.61
960	##	0.016	0.61
961	##	0.016	0.61
962	##	0.016	0.61
963	##	0.016	0.61
964	##	0.016	0.61
965	##	0.016	0.61
966	##	0.016	0.61
967	##	0.016	0.61
968	##	0.016	0.61
969	##	0.016	0.61
970	##	0.017	0.61
971	##	0.017	0.61
972	##	0.017	0.61
973	##	0.017	0.61
974	##	0.017	0.61

975	##	0.017	0.61
976	##	0.017	0.61
977	##	0.017	0.61
978	##	0.017	0.61
979	##	0.017	0.61
980	##	0.017	0.61
981	##	0.017	0.61
982	##	0.017	0.61
983	##	0.017	0.61
984	##	0.017	0.61
985	##	0.017	0.61
986	##	0.017	0.61
987	##	0.017	0.61
988	##	0.017	0.61
989	##	0.017	0.61
990	##	0.017	0.61
991	##	0.017	0.61
992	##	0.017	0.61
993	##	0.017	0.61
994	##	0.017	0.61
995	##	0.017	0.61
996	##	0.017	0.61
997	##	0.017	0.61
998	##	0.017	0.61
999	##	0.017	0.61
1000	##	0.017	0.61
1001	##	0.017	0.61

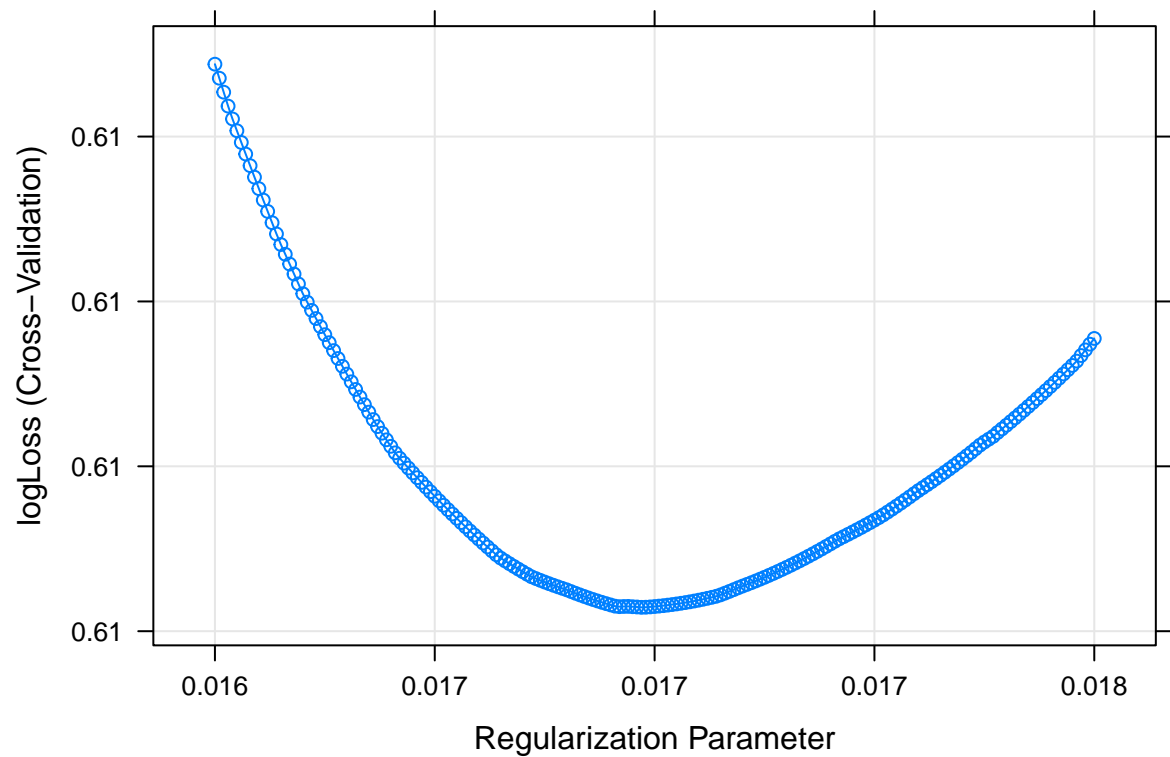
1002	##	0.017	0.61
1003	##	0.017	0.61
1004	##	0.017	0.61
1005	##	0.017	0.61
1006	##	0.017	0.61
1007	##	0.017	0.61
1008	##	0.017	0.61
1009	##	0.017	0.61
1010	##	0.017	0.61
1011	##	0.017	0.61
1012	##	0.017	0.61
1013	##	0.017	0.61
1014	##	0.017	0.61
1015	##	0.017	0.61
1016	##	0.017	0.61
1017	##	0.017	0.61
1018	##	0.017	0.61
1019	##	0.017	0.61
1020	##	0.017	0.61
1021	##	0.017	0.61
1022	##	0.017	0.61
1023	##	0.017	0.61
1024	##	0.017	0.61
1025	##	0.017	0.61
1026	##	0.017	0.61
1027	##	0.017	0.61
1028	##	0.017	0.61

1029	##	0.017	0.61
1030	##	0.017	0.61
1031	##	0.017	0.61
1032	##	0.017	0.61
1033	##	0.017	0.61
1034	##	0.017	0.61
1035	##	0.017	0.61
1036	##	0.017	0.61
1037	##	0.017	0.61
1038	##	0.017	0.61
1039	##	0.017	0.61
1040	##	0.017	0.61
1041	##	0.017	0.61
1042	##	0.017	0.61
1043	##	0.017	0.61
1044	##	0.017	0.61
1045	##	0.017	0.61
1046	##	0.017	0.61
1047	##	0.017	0.61
1048	##	0.017	0.61
1049	##	0.017	0.61
1050	##	0.017	0.61
1051	##	0.017	0.61
1052	##	0.017	0.61
1053	##	0.017	0.61
1054	##	0.017	0.61
1055	##	0.017	0.61

1056	##	0.017	0.61
1057	##	0.017	0.61
1058	##	0.017	0.61
1059	##	0.017	0.61
1060	##	0.017	0.61
1061	##	0.017	0.61
1062	##	0.017	0.61
1063	##	0.017	0.61
1064	##	0.017	0.61
1065	##	0.017	0.61
1066	##	0.017	0.61
1067	##	0.017	0.61
1068	##	0.017	0.61
1069	##	0.017	0.61
1070	##	0.018	0.61
1071	##	0.018	0.61
1072	##	0.018	0.61
1073	##	0.018	0.61
1074	##	0.018	0.61
1075	##	0.018	0.61
1076	##	0.018	0.61
1077	##	0.018	0.61
1078	##	0.018	0.61
1079	##	0.018	0.61
1080	##	0.018	0.61
1081	##	0.018	0.61
1082	##	0.018	0.61

1083	##	0.018	0.61
1084	##	0.018	0.61
1085	##	0.018	0.61
1086	##	0.018	0.61
1087	##	0.018	0.61
1088	##	0.018	0.61
1089	##	0.018	0.61
1090	##	0.018	0.61
1091	##	0.018	0.61
1092	##	0.018	0.61
1093	##	0.018	0.61
1094	##	0.018	0.61
1095	##	0.018	0.61
1096	##	0.018	0.61
1097	##	0.018	0.61
1098	##	0.018	0.61
1099	##	0.018	0.61
1100	##	0.018	0.61
1101	##	0.018	0.61
1102	##	0.018	0.61
1103	##	0.018	0.61
1104	##	0.018	0.61
1105	##	0.018	0.61
1106	##	0.018	0.61
1107	##	0.018	0.61
1108	##	0.018	0.61
1109	##	0.018	0.61

```
1110 ## 0.018 0.61
1111 ## 0.018 0.61
1112 ## 0.018 0.61
1113 ## 0.018 0.61
1114 ## 0.018 0.61
1115 ## 0.018 0.61
1116 ## 0.018 0.61
1117 ## 0.018 0.61
1118 ## 0.018 0.61
1119 ## 0.018 0.61
1120 ## 0.018 0.61
1121 ##
1122 ## Tuning parameter 'alpha' was held constant at a value of 1
1123 ## logLoss was used to select the optimal model using the smallest value.
1124 ## The final values used for the model were alpha = 1 and lambda = 0.017.
```



1126 ## alpha lambda

1127 ## 98 1 0.017

1128 ## [1] 200 2

1129 ## hate nothate

1130 ## 1 0.46 0.54

1131 ## 2 0.69 0.31

1132 ## 3 0.11 0.89

1133 ## 4 0.52 0.48

1134 ## 5 0.67 0.33

1135 ## 6 0.52 0.48

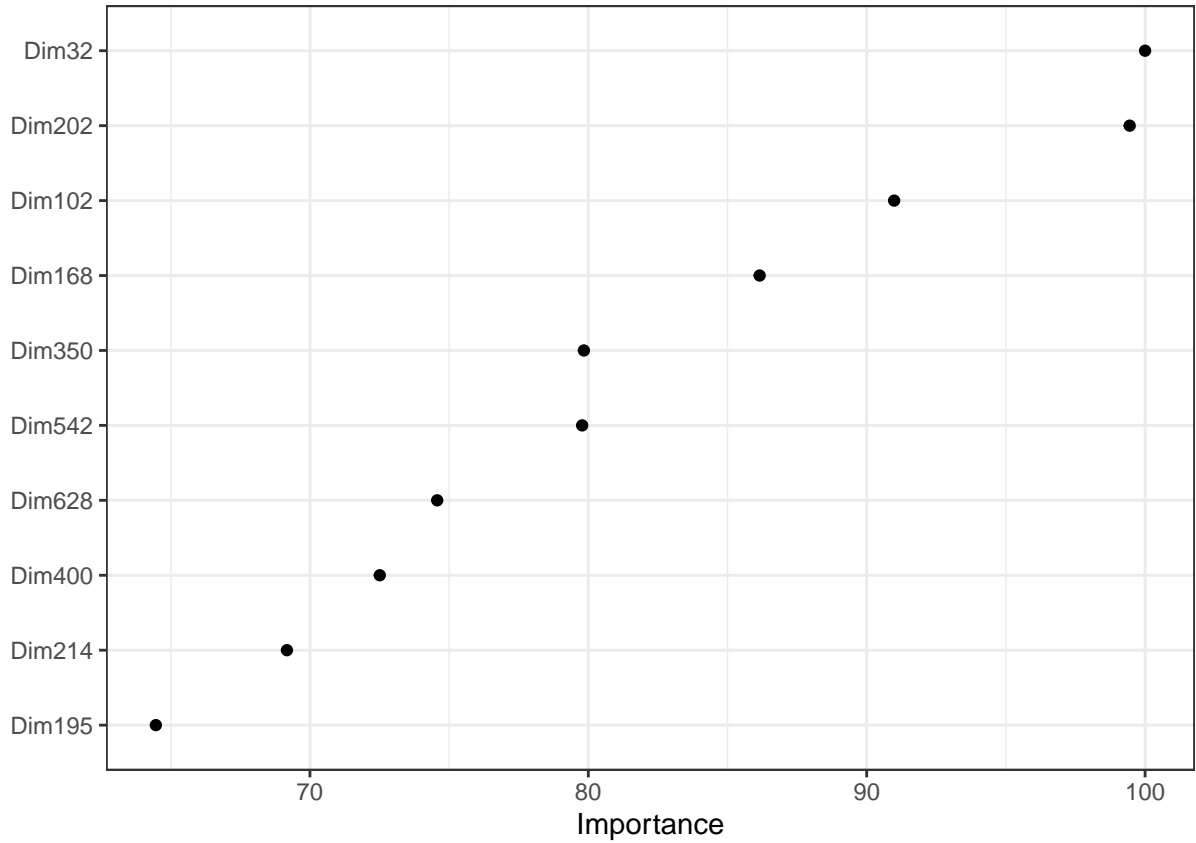
1136 ## Warning: Unknown or uninitialised column: 'hate'.

1137 ## pred_class_12

1138 ## 0 1

1139 ## hate 32 82

1140 ## nothate 50 36



1141

1142	##	[,1]
1143	## (Intercept)	-0.28
1144	## Dim32	-0.22
1145	## Dim202	0.22
1146	## Dim102	-0.20
1147	## Dim168	0.19
1148	## Dim350	0.18
1149	## Dim542	0.18
1150	## Dim628	0.16
1151	## Dim400	-0.16
1152	## Dim214	0.15

References

1153
1154 `## Warning in utils::citation(x[pkg], auto = if (no_citations[pkg]) TRUE else`
1155 `## NULL): no date field in DESCRIPTION file of package 'recipes'`

1156 Davidson, T., Warmley, D., Macy, M., & Weber, I. (2017). Automated hate speech
1157 detection and the problem of offensive language. In *Proceedings of the*
1158 *international AAAI conference on web and social media* (Vol. 11).

1159 Williams, M. L., Burnap, P., Javed, A., Liu, H., & Ozalp, S. (2020). Hate in the
1160 machine: Anti-black and anti-muslim social media posts as predictors of offline
1161 racially and religiously aggravated crime. *The British Journal of Criminology*,
1162 *60*(1), 93–117.

Table 1
*Number of
Hate and no
Hate labels*

Label	n
hate	562
nothate	438

Table 2

Evaluation metrics for the RoBERTa - Layer 12 model

model	-LL	AUC	ACC	TPR	TNR	FPR	PRE
Logistic Regression	16.88	0.54	0.47	0.51	0.43	0.57	0.40
Logistic Regression with Ridge Penalty	0.61	0.73	0.34	0.42	0.28	0.72	0.31
Logistic Regression with Lasso Penalty	0.61	0.71	0.34	0.42	0.28	0.72	0.31