Conceptual and methodological problems with bias detection and avoidance in natural language processing

Alicja Dobrzeniecka

2021-08-02

Contents

Chapter 1

Introduction

Placeholder

Chapter 2

Cosine similarity and bias detection

Placeholder

2.1 Word embeddings

2.2 Cosine similarity and distance

2.3 Cosine distance in a one-class bias detection

2.4 Cosine distance in a multi-class bias detection

2.5 Limitations of the approach

Chapter 3

Walkthrough with the religion dataset

3.1 Loading and understanding the dataset

We start with loading the libraries needed for the analysis.

```
library(ggplot2)
library(ggthemes)
library(rethinking)
library(tidyverse)
library(ggpubr)
library(kableExtra)
library(dplyr)
library(ggExtra)
library(cowplot)
```

We will use the choice of protected words and stereotypical predicates used in Manzini, Lim, Tsvetkov, & Black (2019). This is a decent point of departure, as not only we want to compare our method to that of Manzini, Lim, Tsvetkov, & Black (2019), but also because this data format is fairly general (as contrasted, say, with a set up for binary stereotypes). Note also that the method we develop here can fairly easily be run for different stereotypization patterns. Let's start with explaining the method and its deployment using a dataset obtained for the religion-related protected words.

Let's load, clean a bit and inspect the head of the religion dataset we prepared. In order to obtain this dataset, we calculated the cosine distance between each protected word and each word from both the bias-related attribute groups, which were used in the original study, and to neutral and human control attributes which we added as control groups. For instance, for religion, the bias-related predicates (coming from the original study in Manzini, Lim, Tsvetkov, & Black (2019)) include muslim bias attributes, jew bias attributes, christian bias attributes (see a list in the Appendix).

We decided to add control groups in the form of two classes — neutral words and human-related words. Without a proper control group it is quite hard to compare the resulting cosine distances and decide on their significance in bias detection. We prepared approximately 230 neutral words to double-check the prima-facie neutral hypothesis that their cosine similarity to the protected words will oscillate around 0 (that is, the distances will be around 1). This provides us with a more reliable point of reference. Moreover, we added human attributes that are associated with people in general to investigate whether the smaller cosine distance between protected words and stereotypes can result simply from the fact that the stereotype predicates are associated with humans. For two control groups, we have randomly drawn 230 words that do not express any property usually attributed to humans, and human related attributes.

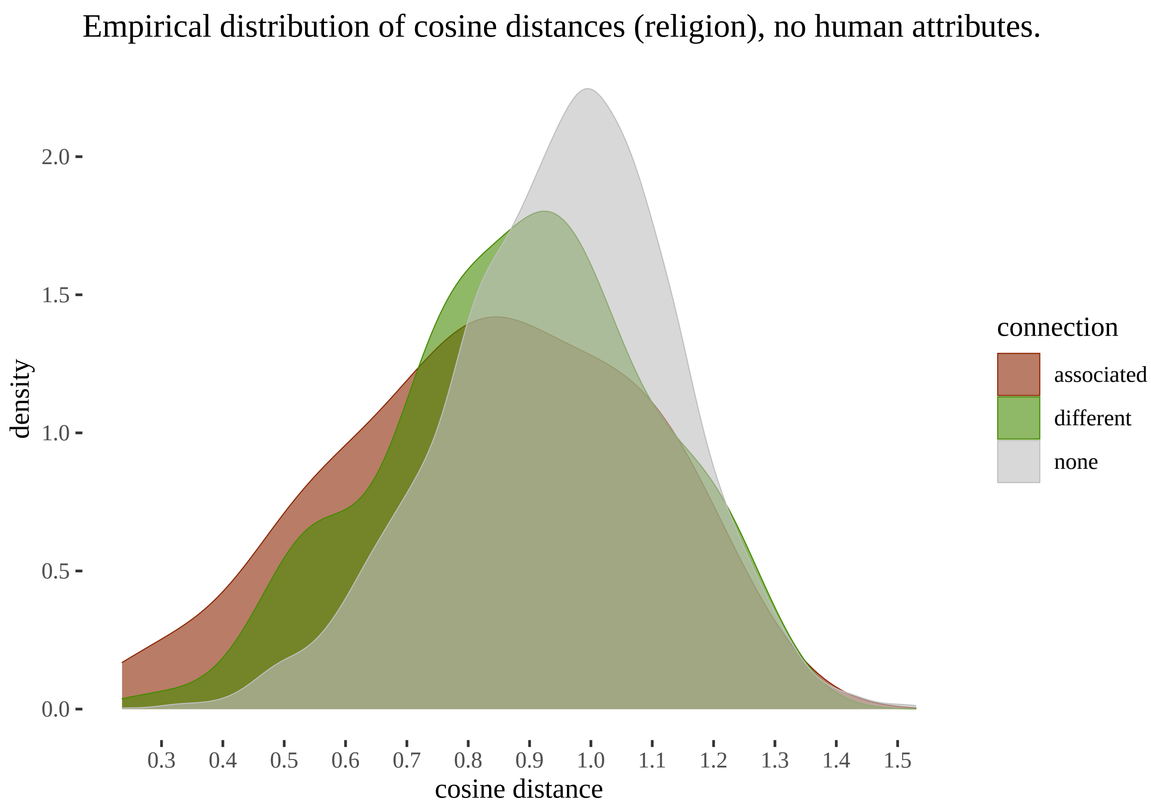
Table 3.1: Head of the religion dataset.

protectedWord	wordToCompare	wordClass	cosineDistance	cosineSimilarity	connection
judaism	violent	muslim	0.7141939	0.2858061	different
judaism	terrorist	muslim	0.7461333	0.2538667	different
judaism	dirty	muslim	1.2002599	-0.2002599	different
judaism	uneducated	muslim	0.7885469	0.2114531	different
judaism	greedy	jewish	1.0026172	-0.0026172	associated
judaism	cheap	jewish	1.2323229	-0.2323229	associated

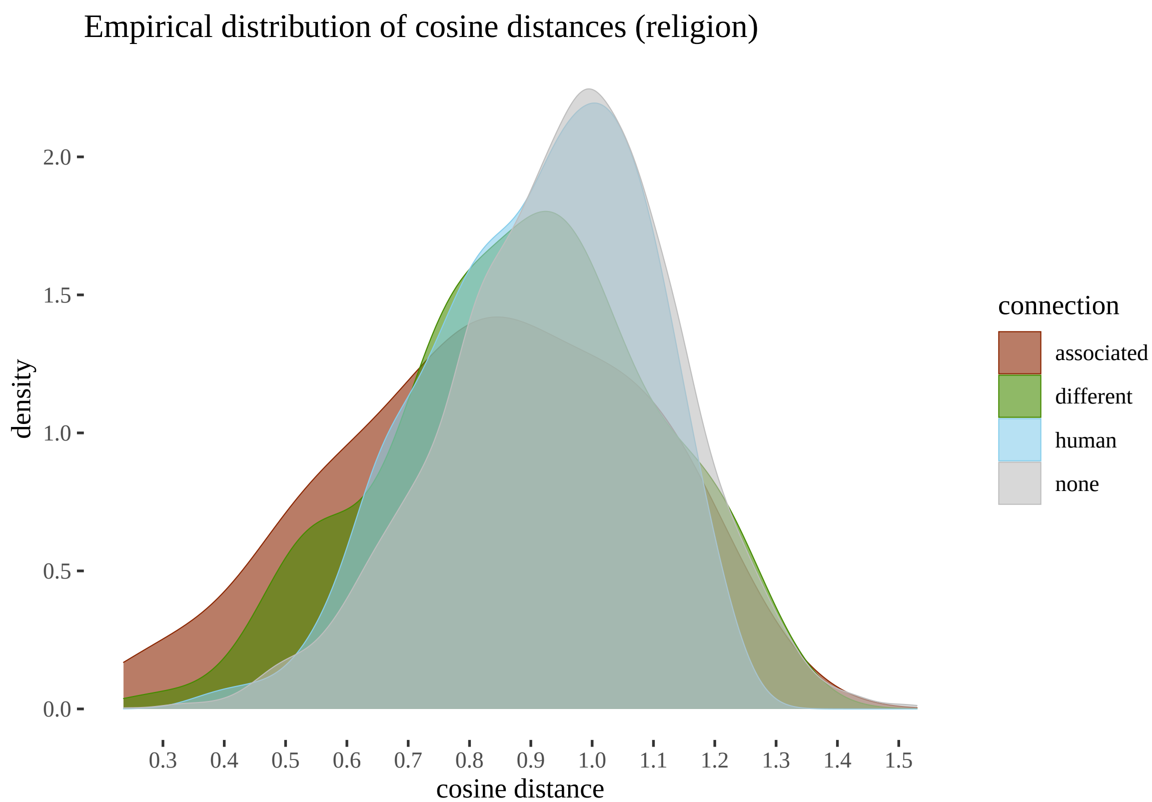
The `protectedWord` column contains words from a protected class that (in a perfect world according to the assumptions of the orignal study) should not be associated with harmful stereotypes. `wordToCompare` contains attributes, including stereotypes and control group words. For each row we compute the cosine distances between a given protected word and a given attribute word. `wordClass` tells us which class an attribute is supposed to be stereotypically associated with, that is, whether the word from `wordToCompare` is associated stereotypically with jews, christians or muslims, or whether it belongs to a control group. `cosineDistance` is simply a calculation of the cosine distance between protected word and atrribute. `cosineSimilarity` contains the result of substracting cosine distance from 1. `connection` contains information about the relation type between a protected word and an attribute. If the attribute is e.g. a harmful jewish stereotype and the protected word is also from the judaism group, the connection has value `associated`. If the attribute is still stereotypically jewish, but the protected word comes from another religion, the connection is labelled as `different`. If the attribute belongs to a neutral group then the connection is labelled as `none` and if an attribute belongs to the `human` class, then the connection is labelled as `human`.

3.2 First look at the empirical distributions

First let's take a look at the empirical distribution of distances by the connection type, initially ignoring the human control class for now.



The first impression is that while there is a shift for associated words towards smaller cosine distances as compared to the neutral words, slightly surprisingly a slightly weaker shift in the same direction is visible for attributes associated with different stereotypes. Moreover, the empirical distributions overlap to a large extent and the means grouped by connection type do not seem too far from each other. In fact, as there is a lot of variety in the cosine distances (as we will soon see), we need to gauge the uncertainty involved, and to look more carefully at individual protected words to get a better idea of how the cosine distance distribution changes for different attribute groups and different protected classes. Now, let's add the human attributes to the picture:



Notice that the distribution for `human` (even though we did our best not to include in it any stereotype-related atributes) is left-skewed, with much overlap with `associated` and `different`, which illustrates the need to take being associated with humans as an important predictor.

Our focus lies in `connection` as a predictor. Morever, later on we'll be interested in looking at the protected words separately, and at protected words split by connection. For technical reasons it is useful to represent these factors as integer vectors.

```
religion$con <- as.integer(religion$connection)
```

```
## Warning: NAs introduced by coercion
```

```
religion$pw <- as.integer(religion$protectedWord)
```

```
## Warning: NAs introduced by coercion
```

```
religion$pwFactor <- factor(paste0(religion$protectedWord, religion$connection))
religion$pwIndex <- as.integer(religion$pwFactor)
```

A short script, `cleanDataset` to make this faster, so equivalently:

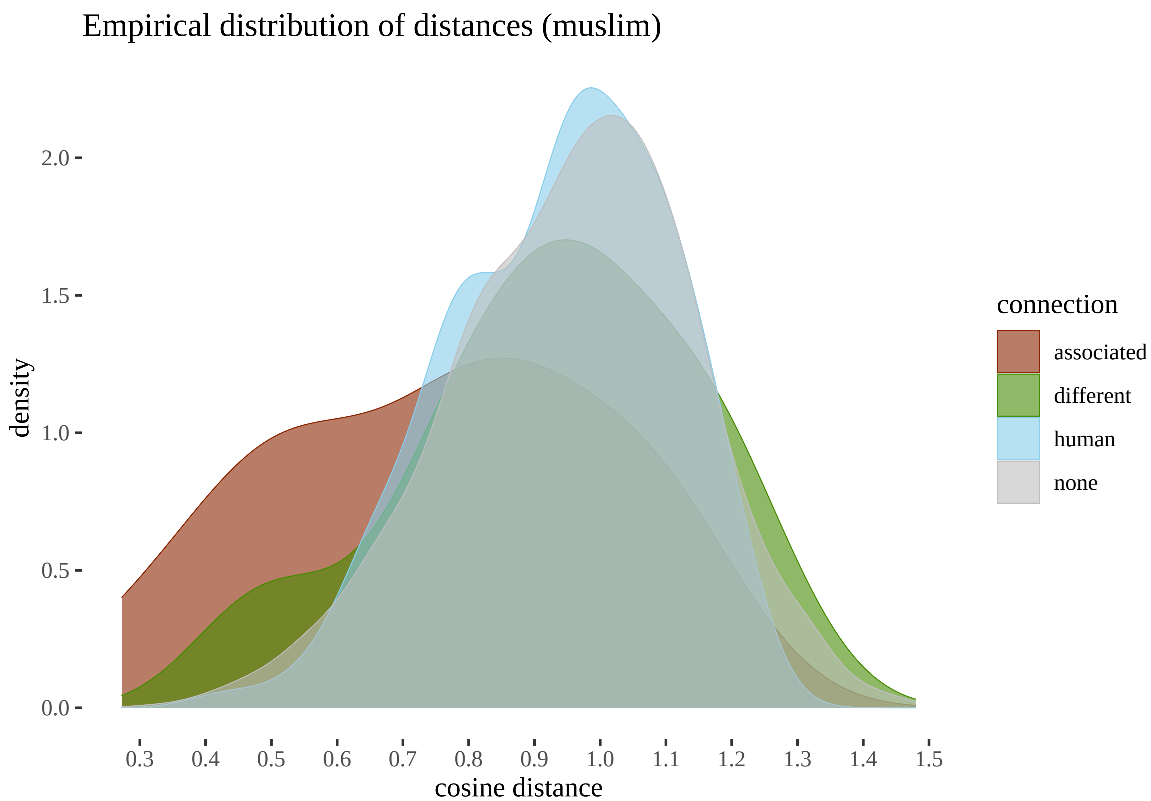
```
source("../functions/cleanDataset.R")
religion <- read.csv("../datasets/religionReddit.csv")[-1]
religion <- cleanDataset(religion,c("christian","human","jewish","muslim","neutral"))
```

```
## Warning in cleanDataset(religion, c("christian", "human", "jewish", "muslim", :
## NAs introduced by coercion

## Warning in cleanDataset(religion, c("christian", "human", "jewish", "muslim", :
## NAs introduced by coercion
```

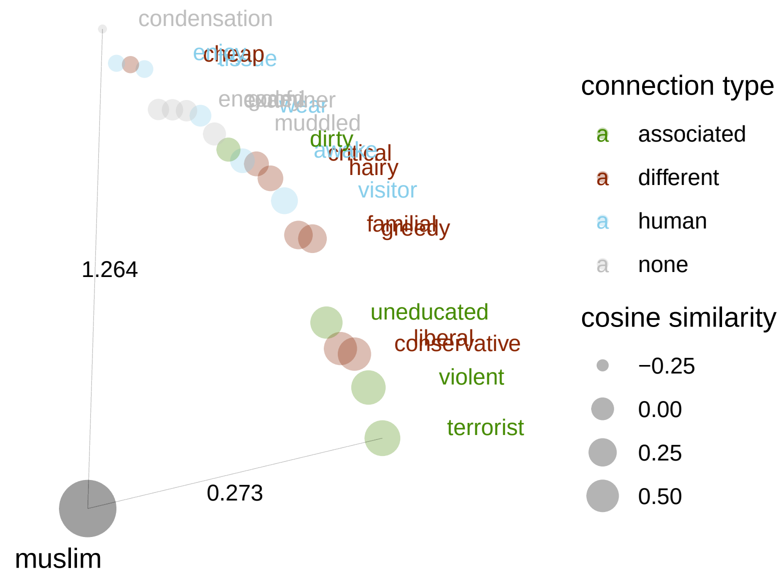
3.3 Looking at the islam-related words

For now, let's focus on five protected words related to islam ("imam," "islam," "mosque," "muslim," and "quran"). The word list associates with islam four stereotypical attributes ("violent," "terrorist," "uneducated" and "dirty"). First, we select and plot the empirical distributions for these protected words.

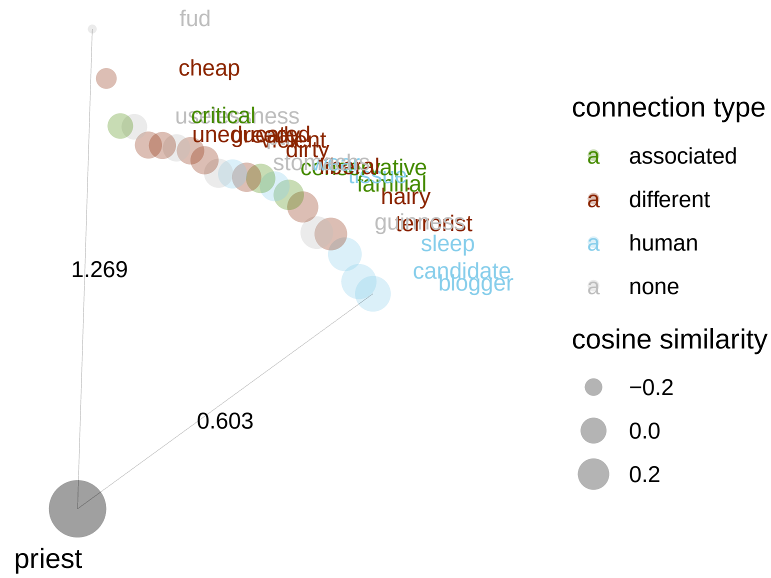


Once we focus on words related to islam, the associated bias seems to be stronger than in the whole dataset. This is a step towards illustrating that the distribution of bias is uneven.

Now, say we want to look at a single protected word. Since the dataset also contains comparison multiple control neutral and human attributes, we randomly select only 5 from `none` and 5 from `human` control groups of those for the visualisation purposes.



Note that the distance between the grey point and the other points is proportional to cosine distance, the non-grey point size is proportional to cosine similarity to the protected word, and color groups by the connection type. So for `muslim` it seems that the stereotypes coming from the word list are fairly well visible. To give you some taste of how uneven the dataset is, compare this to what happens with `priest`.



Here you can see that some human attributes are closer than stereotype attributes, and that there is no clear reason to claim that `associated` attributes are closer than `different` or `human` attributes. This, again, illustrates the need of case-by-case analysis with control groups.

The general idea now is that the word lists provided in different pieces of research are just samples of attributes associates with various stereotypes and should be treated as such: the uncertainty involved and the sample sizes should have clear impact on our estimates.

3.4 Bayesian model structure and assumptions

We will now think of cosine distance as the output variable, and will build a few Bayesian models to compare. First, we just build a baseline model which estimates cosine distance to the attributes separately for each protected word. The underlying idea is that different protected words might in general have different relations to all the attributes and these relations should be our point of departure.[1]

Here is the intuition behind the mathematical Bayesian model involved. Our outcome variable is `cosine difference`, which we take to me normally distributed around the predicted mean for a given protected word (that is, we assume the residuals are normally distributed). The simplest model specification is:

$$cosineDistance_i \sim dnorm(\mu_i, \sigma) \tag{3.1}$$

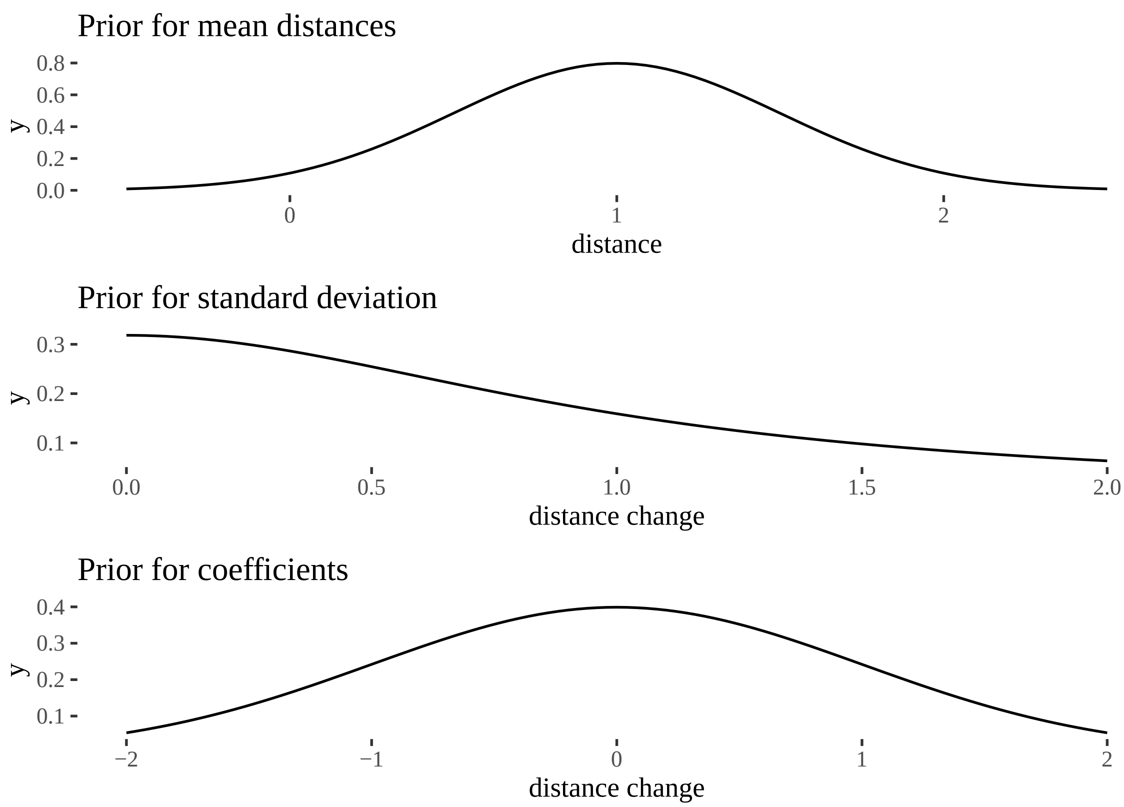
$$\mu_i = m_{pw} \tag{3.2}$$

$$m_{pw}\ dnorm(1, .5) \tag{3.3}$$

$$\sigma \sim dcauchy(0, 1) \tag{3.4}$$

That is, we assume the estimated means might be different for diferent protected words and our prior for the mean and the overal standard deviation are normal with mean 1 and sd=.5 and half-cauchy with parameters `0,1`. Further on we'll also estimate additional impact the connection type may have. For this impact we take a slightly skeptical prior centered around 0 distributed normally with sd = 1. These are fairly weak and slightly skeptical regularizing priors, which can be illustrated as follows:

[1] The construction of the Bayesian models and code for visualisations is due to Rafal Urbaniak.



3.5 Choosing predictors

Now we can define and compile the baseline model. Its parameters will have a posterior distribution obtained using either Hamiltionian Monte Carlo methods (STAN) available through the `rethinking` package.

```
library(rethinking)
options(buildtools.check = function(action) TRUE )
religionBaseline <- ulam(
  alist(
    cosineDistance ~ dnorm(mu,sigma),
    mu <- m[pw],
    m[pw] ~ dnorm(1,.5),
    sigma ~ dcauchy(0,1)
```

```
  ),
  data = religion,
  chains=2 , iter=4000 , warmup=1000,
  start= list(mu = 1, co = 0, sigma= .3),
  log_lik = TRUE, cores=4
)
#saving
#saveRDS(religionBaseline,
#file = "cosineAnalysis/models/religionBaseline.rds")
```

The only reason we need it is the evaluation of connection as a predictor. Does including it in o the model improve the situation? To investigate this, let's now build a model according to the following specification:

$$cosineDistance_i \sim dnorm(\mu_i, \sigma) \tag{3.5}$$

$$\mu_i = m_{pw} + co_{con} \tag{3.6}$$

$$m_{pw}\ dnorm(1, .5) \tag{3.7}$$

$$co_{con}\ dnorm(0, 1) \tag{3.8}$$

$$\sigma \sim dcauchy(0, 1) \tag{3.9}$$

The idea now is that each connection type comes with its own coefficient *co* that has impact on mean distances for protected words taken separately.

```
library(rethinking)
options(buildtools.check = function(action) TRUE )
religionCoefs <- ulam(
  alist(
    cosineDistance ~ dnorm(mu,sigma),
```

```
    mu <- m[pw] + co[con],
    m[pw] ~ dnorm(1,.5),
    co[con] ~dnorm(0,.5),
    sigma ~ dcauchy(0,1)
  ),
  data = religion,
  chains=2 , iter=8000 , warmup=1000,
  log_lik = TRUE
)
```

First, let's see if this model is really better in terms of the Widely Acceptable Information Criterion (WAIC):

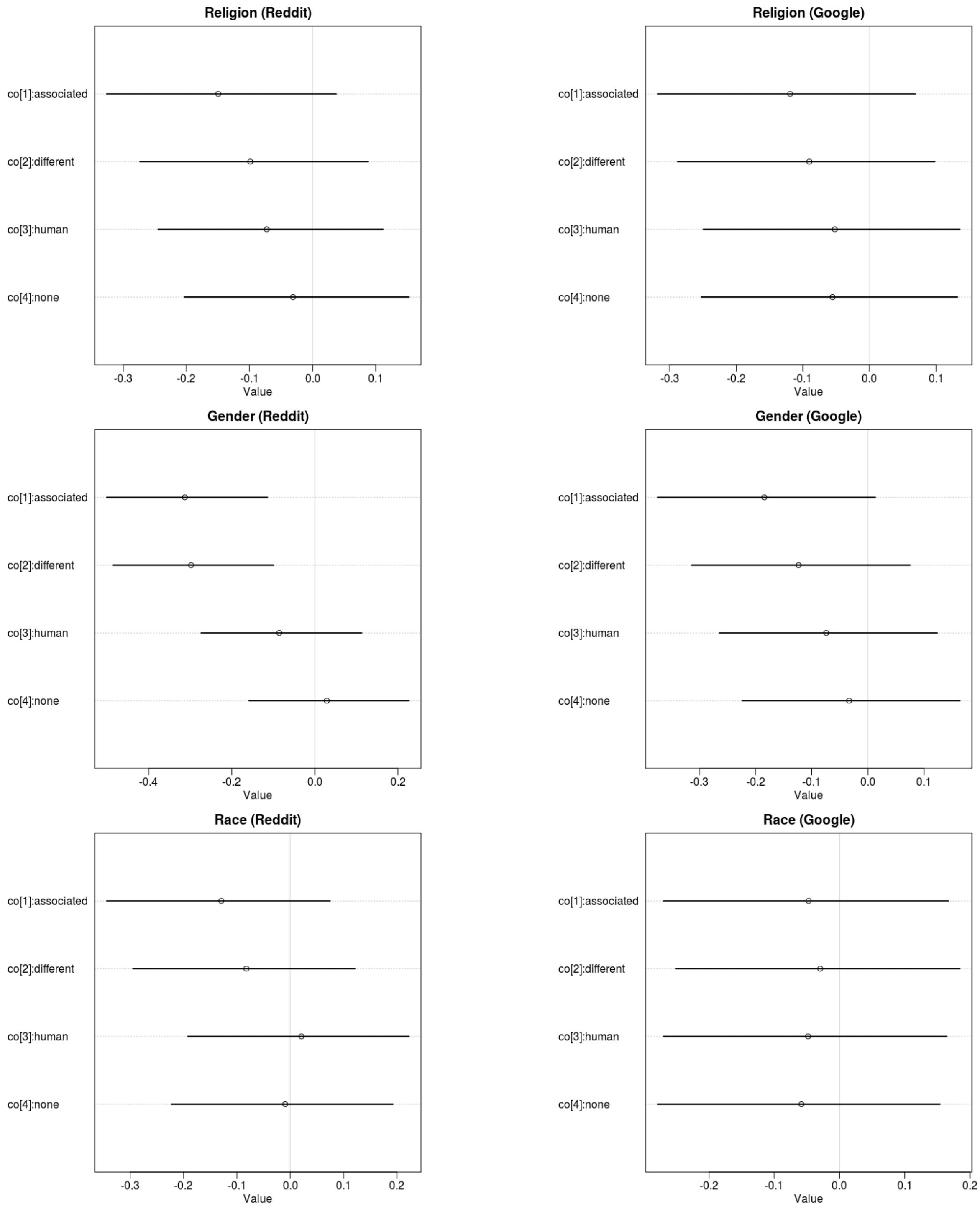
```
##                   WAIC SE dWAIC dSE pWAIC weight
## religionCoefs    -2328 93     0  NA    20      1
## religionBaseline -2283 95    45  17    16      0
```

Clearly, it should be given weight 1 as compared to the baseline model. So far, we've learned that the connection type actually has predictive value. Let's take a look at the coefficient estimates:

```
##              mean        sd       5.5%      94.5%    n_eff    Rhat4
## co[1] -0.14956420 0.1151675 -0.3261650 0.03741930 294.2033 1.001449
## co[2] -0.09880543 0.1145024 -0.2736985 0.08813271 291.5044 1.001564
## co[3] -0.07282752 0.1133894 -0.2447778 0.11158986 287.7820 1.001627
## co[4] -0.03103179 0.1131420 -0.2034442 0.15268770 286.8283 1.001606
```

3.6 Dataset-level coefficients

Let's plot them together with their highest posterior density invervals, for the three topic groups.


Religion (Reddit)
Religion (Google)
Gender (Reddit)
Gender (Google)
Race (Reddit)
Race (Google)
co[1]:associated
co[2]:different
co[3]:human
co[4]:none
Value
-0.4
-0.3
-0.2
-0.1
0.0
0.1
0.2

What should strike us is that while the mean estimates of the coefficients indeed do differ a bit, usually the highest posterior density invervals all include zero, and so we do not have strong reasons to say that, say, as far as the whole religion dataset is involved, being associated indeed is connected with lower cosine distance. A second striking observation is that the estimated impact for associated stereotypes is quite often not too different from the estimated impact of attributes associated with different stereotypes, and both are sometimes not too far from the estimated impact for simply human attributes. In general, once the uncertainty involved is taken seriously by using control groups and statistical uncertainty estimation that does not dispose of pointwise data, the picture which focuses only on differences between means of means is too simplistic.

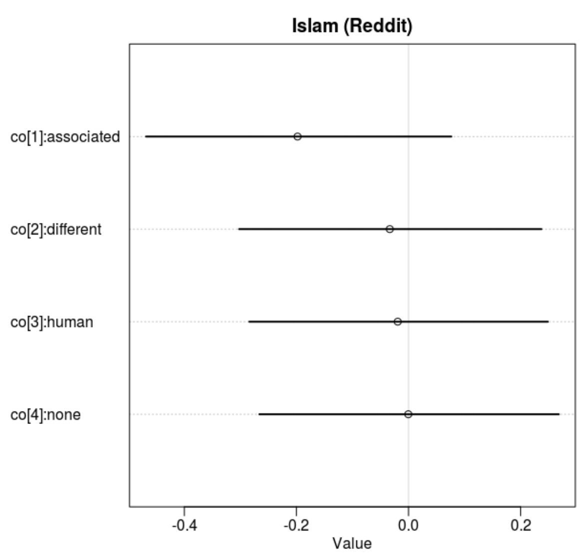
But this doesn't mean important differences for some protected words are not there. For one thing, if you start with a word list that is very uneven, the actually not so bad status of some of the protected words might mask a pretty bad situation in which some other protected words are. For comparison, let's see what a model focused on words related to islam tells us.

```
#this is how we build the model
religion <- read.csv("cosineAnalysis/datasets/religionReddit.csv")[-1]
colnames(religion) <- c("protectedWord","wordToCompare","wordClass",
                        "cosineDistance","cosineSimilarity","connection")
levels(religion$wordClass) <- c("christian","human","jewish","muslim","neutral")
muslimWords <- c("imam","islam","mosque","muslim","quran")
muslim <- religion %>% filter(protectedWord %in% muslimWords)
muslim$protectedWord <- droplevels(muslim$protectedWord)
muslim$pw <- as.integer(muslim$protectedWord)
muslim$con <- as.integer(muslim$connection)
muslim$pwFactor <- factor(paste0(muslim$protectedWord, muslim$connection))
muslim$pwIndex <- as.integer(muslim$pwFactor)
```

```
islamCoefs <- ulam(
  alist(
    cosineDistance ~ dnorm(mu,sigma),
    mu <- m[pw] + co[con],
    m[pw] ~ dnorm(1,.5),
    co[con] ~dnorm(0,.5),
    sigma ~ dcauchy(0,1)
  ),
  data = muslim,
  chains=2 , iter=10000 , warmup=1000, cores = 4,
  log_lik = TRUE
)
```

Let's take a look at the coefficients:

```
##                 mean        sd       5.5%      94.5%    n_eff    Rhat4
## co[1] -0.1979930035 0.1708785 -0.4682696 0.07634977 1789.894 1.003445
## co[2] -0.0334215769 0.1687587 -0.3021954 0.23720938 1738.720 1.003575
## co[3] -0.0192492753 0.1675754 -0.2840596 0.24860974 1732.907 1.003755
## co[4] -0.0003911363 0.1670610 -0.2661047 0.26815172 1723.758 1.003837
```



While muslim words were unusual in the sense that the disparity between associated attributes and others is stronger, the evidence is still not conclusive. This is because the variation even within islam-related words is large enough (and sample sizes sufficiently small) for all the highest posterior density invervals to still include zeros.

So, it seems, taking a closer look does seem to make a difference. The question is, what happens if we do take a close look at the level of protected words?

Chapter 4

Protected-word level analysis

4.1 Model structure and assumptions

Let's turn then to data analysis that takes a look at protected words separately. This time, for each combination of a protected word and a connection status we will have a separate mean cosine distance estimate, each coming with its own highest posterior density interval. This means we will use indices that are result from all such combinations (and then we will split them up in the model precis to build visualisation, feel free to look at the `visualiseStats.R` script for details).

```
options(buildtools.check = function(action) TRUE ) #removes install pop-up request
religion$pwFactor <- factor(paste0(religion$protectedWord, "-", religion$connection))
religion$pwIndex <- as.integer(religion$pwFactor)

religionSeparate <- ulam(
  alist(
    cosineDistance ~ dnorm(mu,sigma),
    mu <- c[pwIndex],
    c[pwIndex] ~ dnorm(1,.5),
```

```
    sigma ~ dcauchy(0,1)
  ),
  data = religion,
  chains=2 , iter=10000 , warmup=1000,
  start=list(no = 1, a = 0, d = 0, sigma= .3), log_lik =  TRUE
)
```

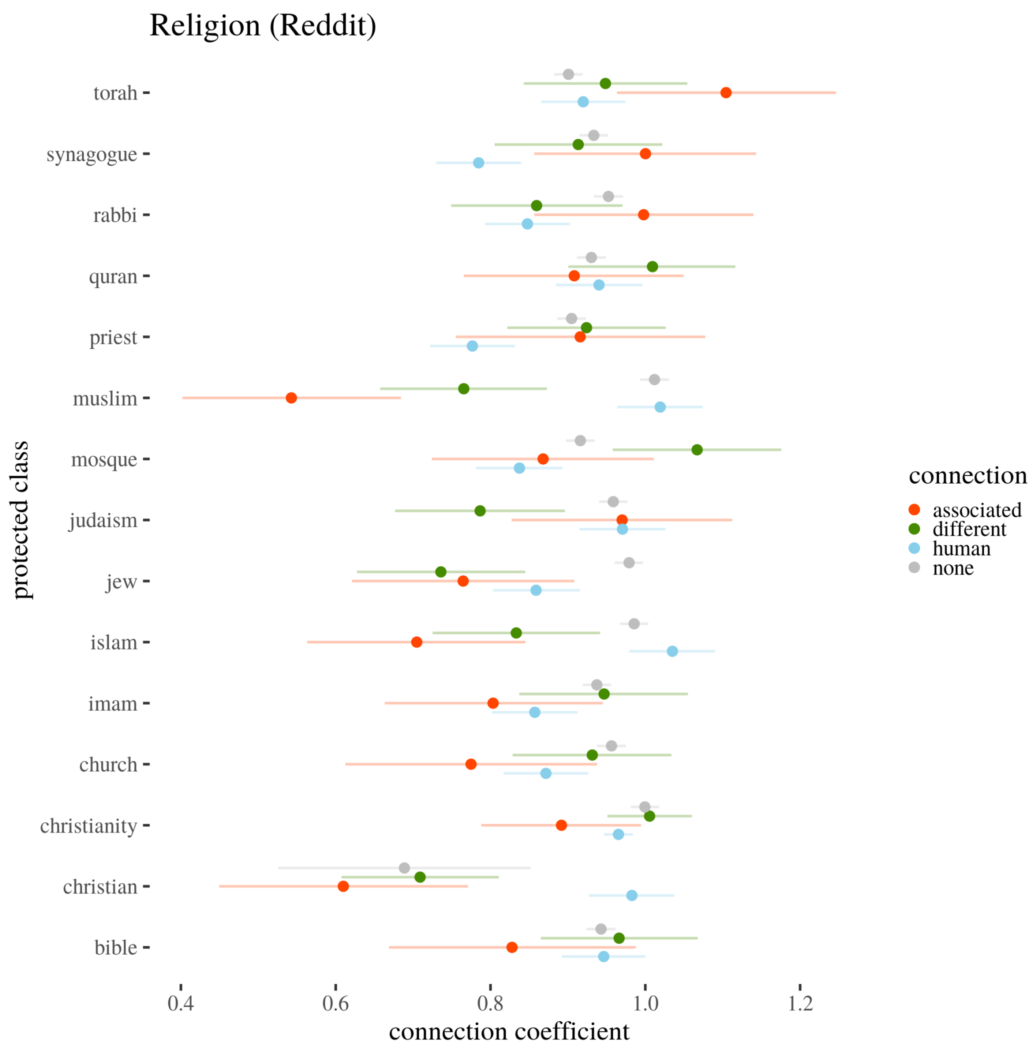
Let's see if the individualized model does better than the previous models in light of WAIC which does add penalty for the number of parameters.

```
compareBaselineCoefsSeparate<- readRDS("../datasets/compareBaselineCoefsSeparate.rds")
compareBaselineCoefsSeparate
```

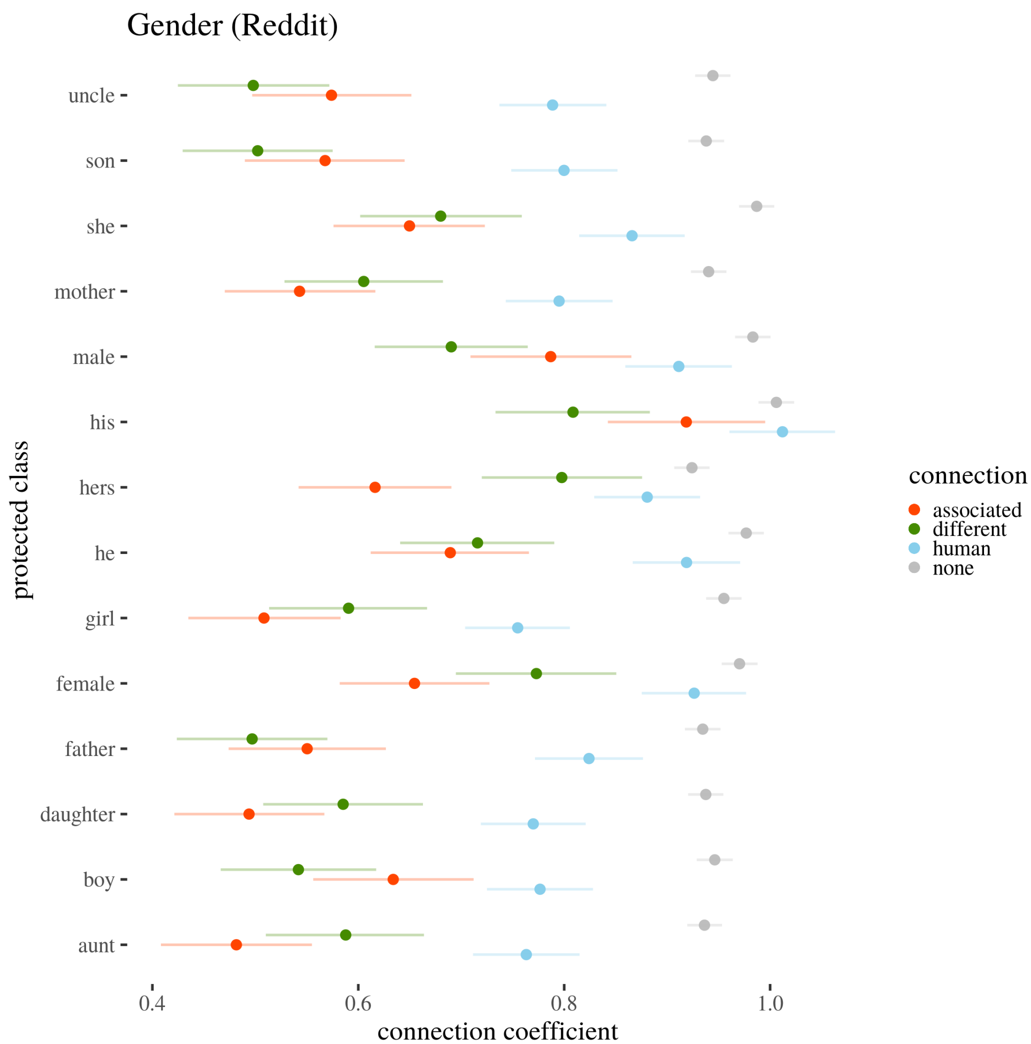
```
##                    WAIC SE dWAIC dSE pWAIC weight
## religionSeparate -2400 93     0  NA    60      1
## religionCoefs    -2328 93    72  29    20      0
## religionBaseline -2283 95   117  37    16      0
```

4.2 Protected classes in Reddit and Google embeddings

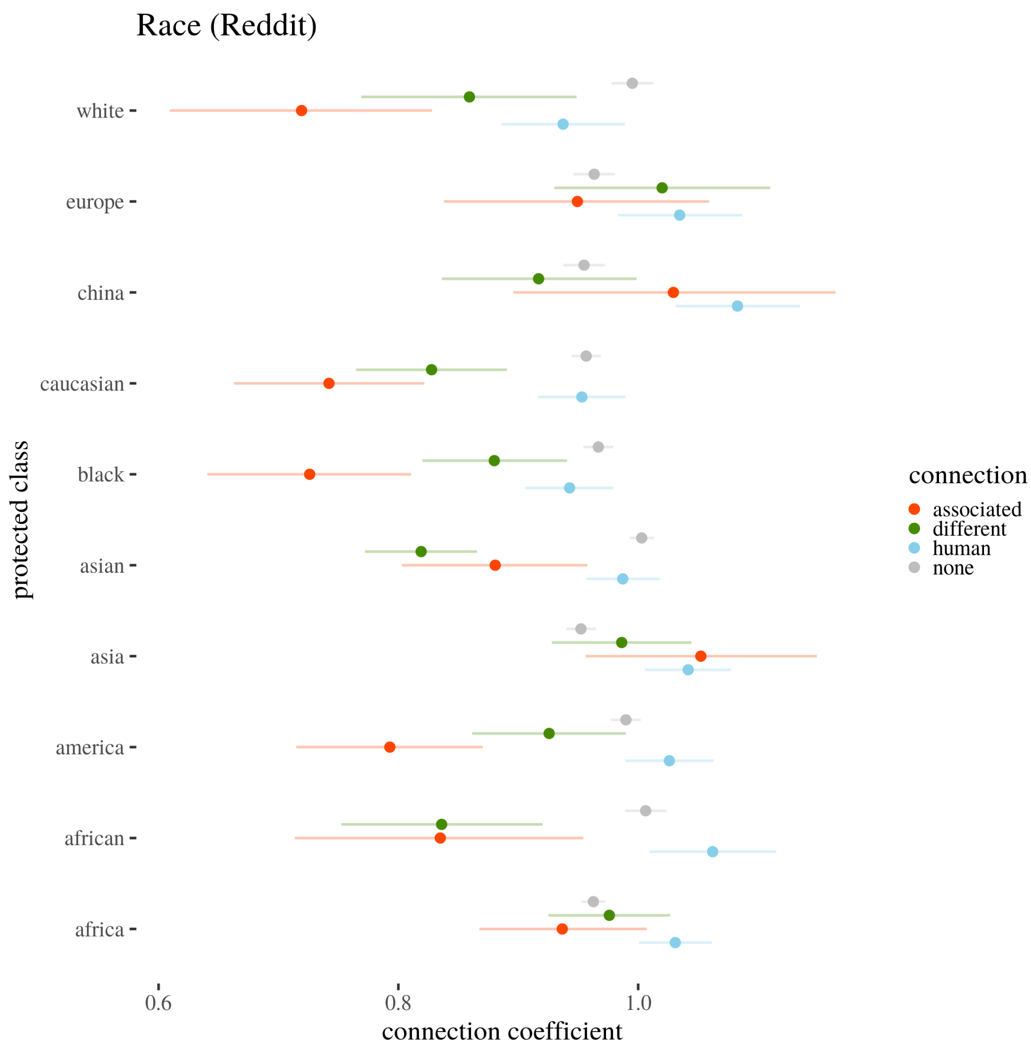
It seems that we do want to prefer this model, despite its relative complication. Now, what does it tell us about the protected words? Let's visualise the predicted means together with 89% highest posterior density intervals.



Before we move on, let's perform analogous analyses for the remaining types of supposed bias: gender and race (the model building is analogous).



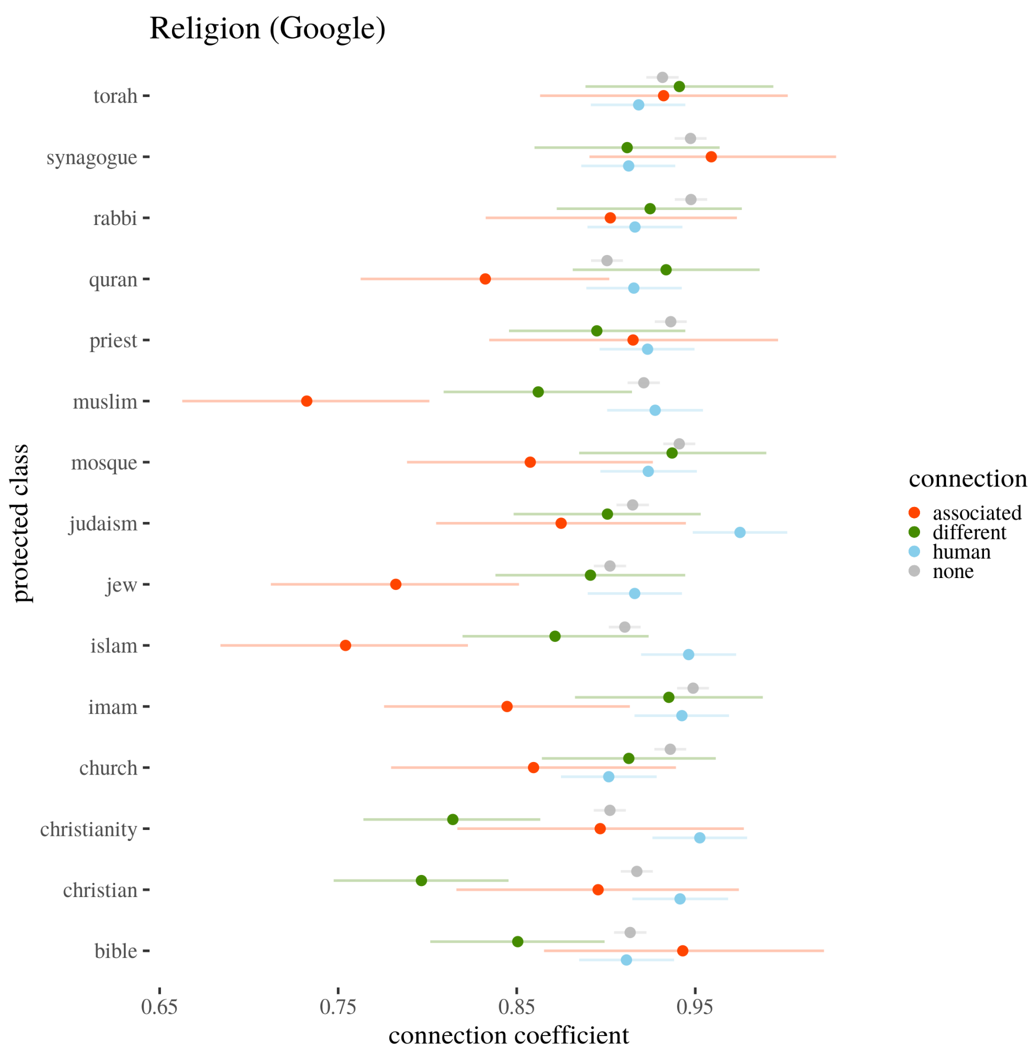
Gender (Reddit)
uncle
son
she
mother
male
his
hers
he
girl
female
father
daughter
boy
aunt
protected class
0.4
0.6
0.8
1.0
connection coefficient
connection
associated
different
human
none

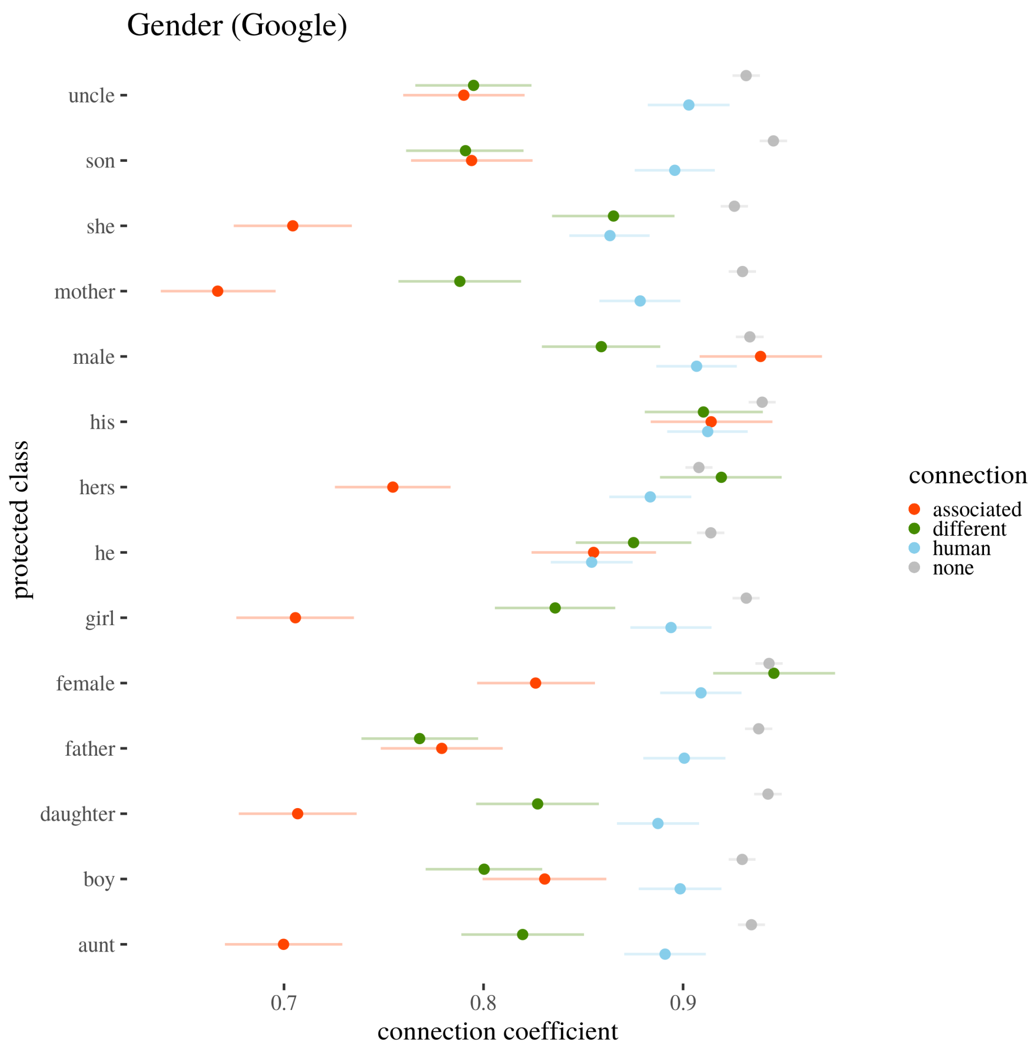


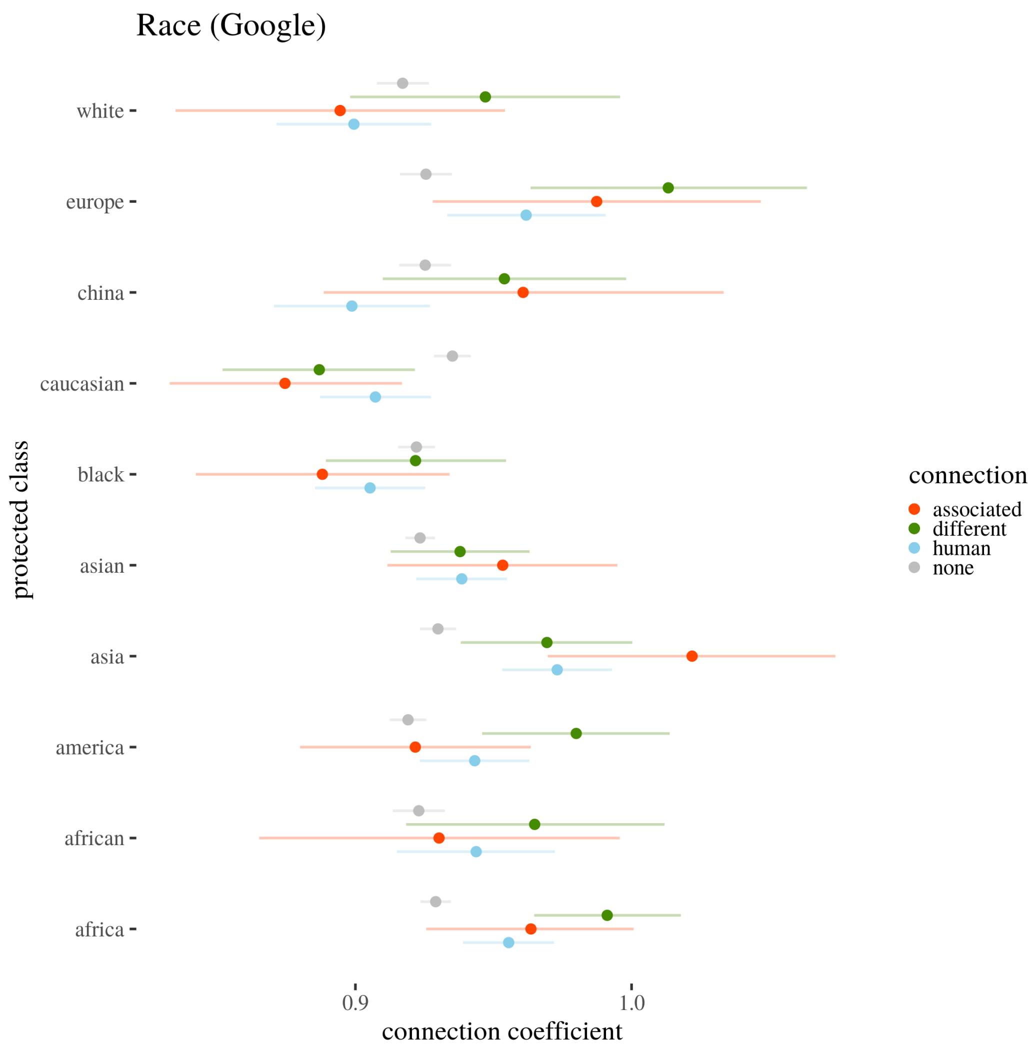
We first encountered the GoogleNews word embedding model in Bolukbasi, Chang, Zou, Saligrama, & Kalai (2016). They use it for their calculations as written in Tolga Bolukbasi code[1]. We decided to compare the results obtained with the use of Reddit L2 model and the ones that we got by GoogleNews model. The details of the model can be found here Google News model. One of the main differences between the models is that Reddit word embeggings have 50 dimensions and GoogleNews word embeddings have 300 dimensions. As the dimension increases, the vectors can capture much more

[1]https://github.com/tolga-b/debiaswe

information, but this is not always the case. According to some researchers, the amount of dimensions should be chosen by taking into account the type of corpus and its features. In our case both Reddit and GoogleNews models are already trained and ready to use so we do not analyse further the choice of their hyperparameters.




Gender (Google)
uncle
son
she
mother
male
his
hers
he
girl
female
father
daughter
boy
aunt
protected class
0.7
0.8
0.9
connection coefficient
connection
associated
different
human
none



We leave a discussion of these results for later, for now let us also inspect the impact of debiasing.

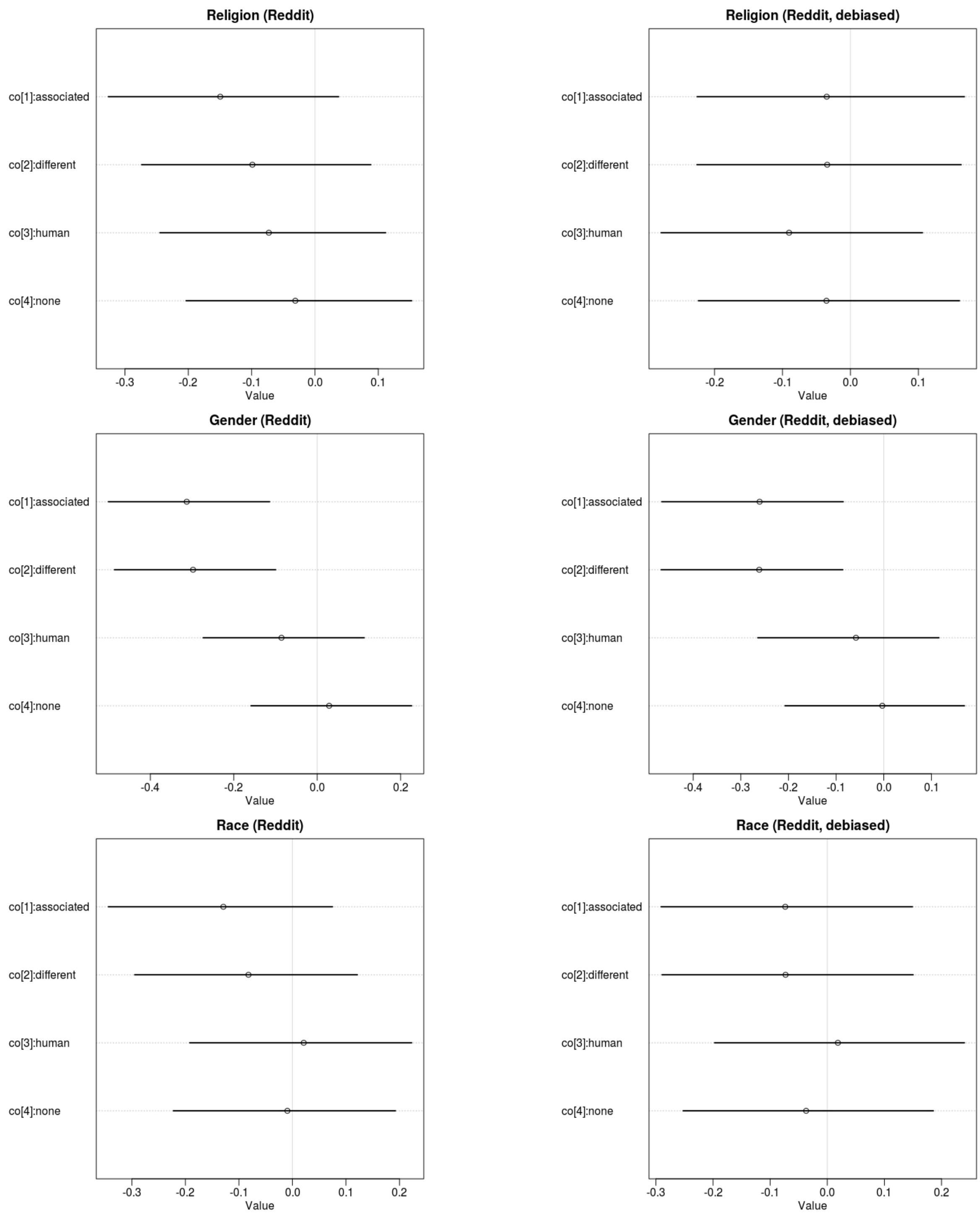
Chapter 5

The role of debiasing

We also applied to the word embeddings hard debiasing method found in Manzini, Lim, Tsvetkov, & Black (2019). As it was mentioned before, debiasing consists of two components: identifying the bias subspace and then removing this vector subspace from the chosen embeddings. The aim is to increase the cosine distance between protected words and the set of attributes. After debiasing the Reddit word embeddings we calculated the cosine distance again to see if there are any significant changes in terms of the similarity between protected words and the attributes.

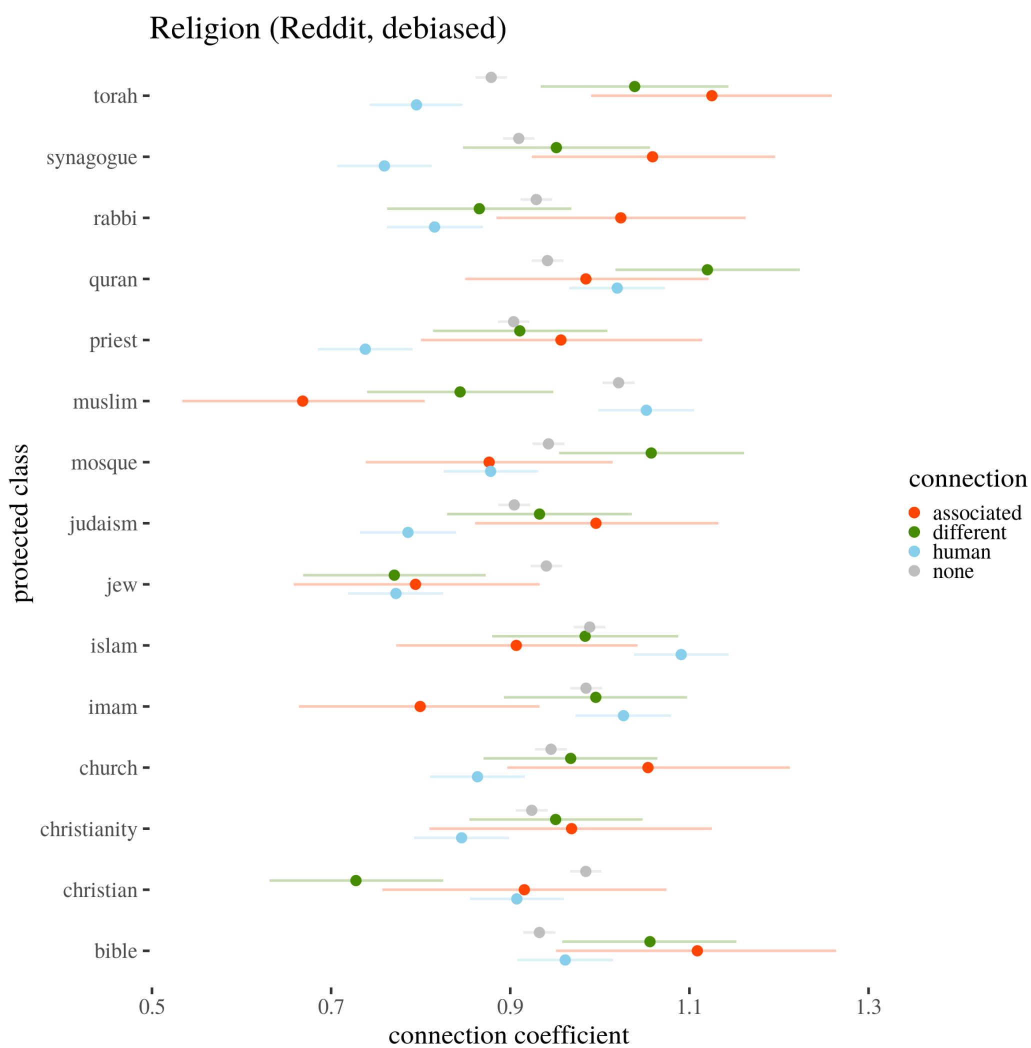
5.1 Dataset-level coefficients after debiasing

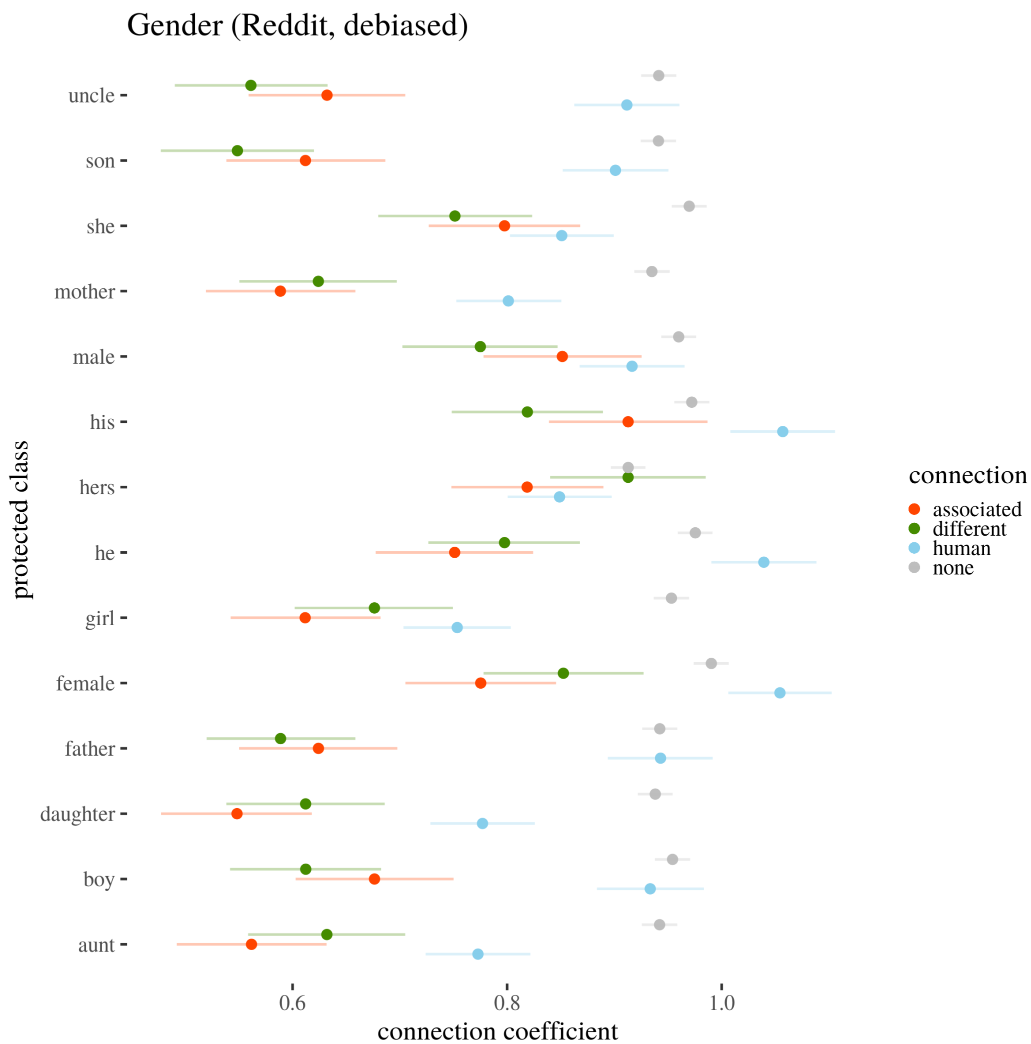
First, let's look at coefficient estimated for the whole datasets, as compared to their estimation prior to debiasing. One may observe very slight changes in the estimated coefficients. It seems that in religion dataset the changes are the most significant as the mean moves towards zero which stands for no similarity between words.

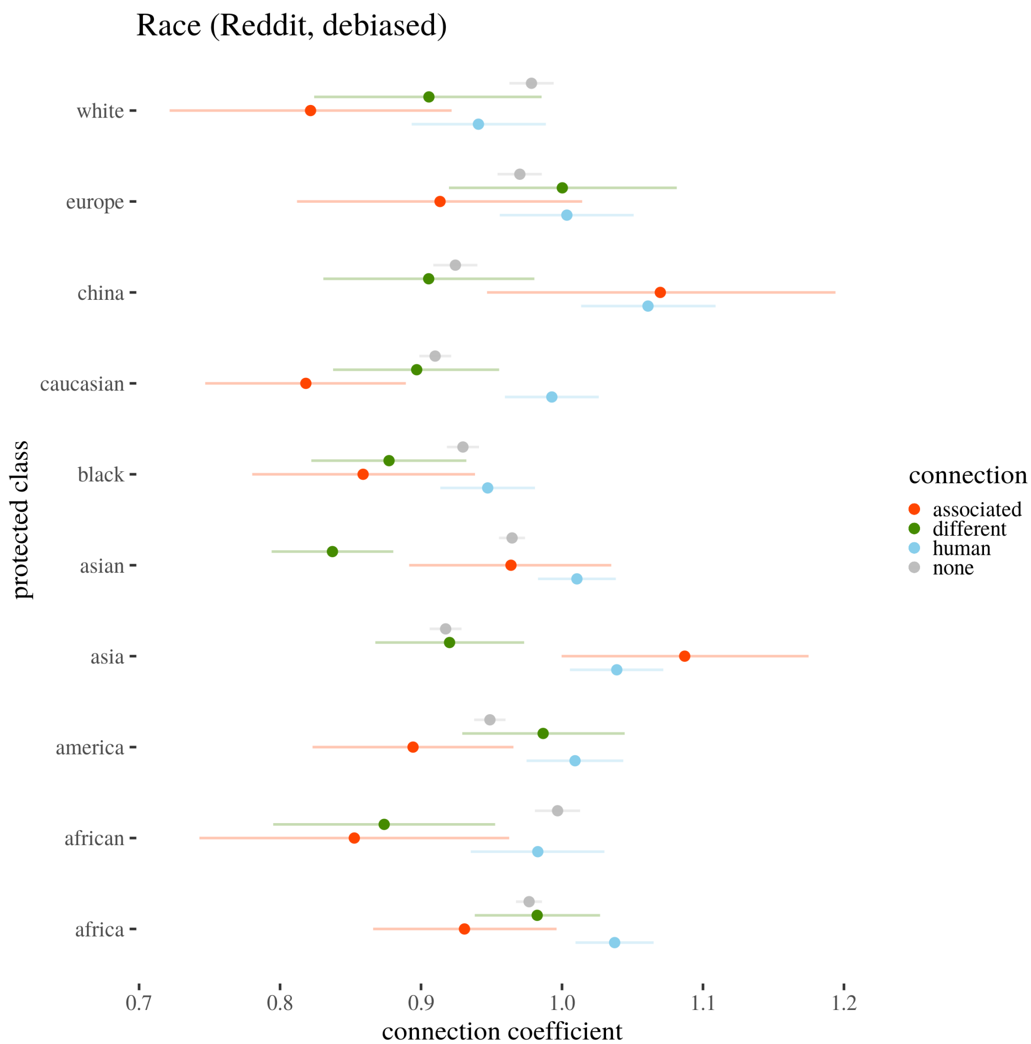

Religion (Reddit)
Religion (Reddit, debiased)
Gender (Reddit)
Gender (Reddit, debiased)
Race (Reddit)
Race (Reddit, debiased)
co[1]:associated
co[2]:different
co[3]:human
co[4]:none
Value
-0.4
-0.3
-0.2
-0.1
0.0
0.1
0.2

5.2 Protected classes after debiasing

Now, perhaps, the effects of debiasing will be better appreciated if we look at the level of protected words. After all, the hope is, the situation of extremely ill-positioned protected words have improved?




Gender (Reddit, debiased)
uncle
son
she
mother
male
his
hers
he
girl
female
father
daughter
boy
aunt
protected class
0.6
0.8
1.0
connection coefficient
connection
associated
different
human
none



One may notice that in a religion data set most of the distances for `associated` words slightly increased. Some of the distances from `different` class are still smaller than for the `associated` class. The distances for all classes seem in most cases to cluster together even after debiasing. The reason may be that uncertainty is quite high in `associated` and `different` class because the number of samples is so small. Additionally in some cases distances from `human` class are smaller than for the `assosiated` class which also makes it less clear how to interpret the results, if the distances are really able to catch the bias presence.

It is worth paying attention to the gender debiased dataset. The change is really minor, still with zero out of the HPDI range. The `different` attributes still have high similarity value. It seems that the distances for gender protected words are still clustered together which suggests that this method is unable to catch the complex bias presence. One could conclude that co-occurrence of certain terms does not always mean direct connection in terms of terms associations. The fact that some of stereotypically associated with female attributes still have high similarity with male protected words suggest that other techniques should be applied to detect and remove the bias.

At the same time, there is a small improvement in race dataset. In some cases the distances just cluster together. In general distances for `assosiated` class are still higher than those for the rest of classes. One may assume that there reason for so small change is that either the metric does not indicate properly biases or that the issue is more complex and subtracting subspaces is not enough.

Chapter 6

Discussion and summary

Placeholder

Appendix

Placeholder

Original wordlist from (Manzini, Lim, Tsvetkov, & Black, 2019)

Religion

Gender

Race

Our control groups

Human neutral attributes

Non-human neutral attributes

Bolukbasi, T., Chang, K.-W., Zou, J. Y., Saligrama, V., & Kalai, A. (2016). Man is to computer programmer as woman is to homemaker? Debiasing word embeddings. *CoRR*, *abs/1607.06520*. Retrieved from http://arxiv.org/abs/1607.06520

Manzini, T., Lim, Y. C., Tsvetkov, Y., & Black, A. W. (2019). Black is to criminal as caucasian is to police: Detecting and removing multiclass bias in word embeddings. Retrieved from http://arxiv.org/abs/1904.04047