

weat.R

```
source('word_utils.R')

####WEAT METHOD function (method used for first half of tests)
weat <- function(words_df, target1, target2, att1, att2) {
  #' This function will return test statistic and effect size computed using WEAT method
  #' @param words_df dataframe holding word vectors where first column is the word itself
  #' @param target1 first vector of target words
  #' @param target2 second vector of target words
  #' @param att1 first vector of attribute words
  #' @param att2 second vector of attribute words

  #get test statistic and effect size from nested function "test_weat()"
  print("Computing test statistic and effect size...")
  result <- .compute_weat(words_df, target1, target2, att1, att2)
  #store test statistic
  t <- result[1]
  effect <- result[2]

  #estimate the mean/variance of test statistic distribution for calculating p-value
  #calculate two-sided p-value
  print("Computing p-value...")
  p <- .compute_p_value(words_df, t, target1, target2, att1, att2)

  #return p-value and effect size
  return(c(p, effect))
}

.compute_p_value <- function(words_df, t, target1, target2, att1, att2) {
  #' Estimates the t-distribution and computes a two-sided p-value for test statistic t
  #' @param words_df dataframe holding word vectors where first column is the word itself
  #' @param t test statistic/numeric value
  #' @param target1 first vector of target words
  #' @param target2 second vector of target words
  #' @param att1 first vector of attribute words
  #' @param att2 second vector of attribute words

  targets <- c(target1, target2)
  ts <- c()
  # run multiple times to estimate test statistic distribution (keeping at 20 for testing purposes)
  for (i in 1:20) {
    # randomize the target words into two groups
    target1 <- sample(targets, length(targets)/2, replace = FALSE)
    target2 <- setdiff(targets, target1)
    # compute t
    test <- .compute_weat(words_df, target1, target2, att1, att2)[1]
    ts[i] <- test
  }
  # mean/variance of test statistic distribution
  mn_t <- mean(ts)
  var_t <- var(ts)
}
```

```

# two-sided p-value
p <- pnorm(t, mean = mn_t, sd = sqrt(var_t), lower.tail = FALSE)

return(p)
}

.compute_weat <- function(words_df, target1, target2, att1, att2) {
  #' Use WEAT method return test statistic and effect size between two sets of target and
  #' attribute words
  #' @param words_df dataframe holding word vectors where first column is the word itself
  #' @param target1 first vector of target words
  #' @param target2 second vector of target words
  #' @param att1 first vector of attribute words
  #' @param att2 second vector of attribute words

  # Return matrices of word vectors for each set of words
  t1_mat <- get_word_vectors(words_df, target1)
  t2_mat <- get_word_vectors(words_df, target2)
  a1_mat <- get_word_vectors(words_df, att1)
  a2_mat <- get_word_vectors(words_df, att2)

  # EFFECT SIZE
  t_effect <- .calc_t_and_effect_size(t1_mat, t2_mat, a1_mat, a2_mat)

  # return test statistic and effect size
  return(t_effect)
}

.calc_t_and_effect_size <- function(t1_mat, t2_mat, a1_mat, a2_mat) {
  #' Returns the test statistic
  #' i.e. a measure of the differential association of the two sets of target words
  #' with the attribute
  #' @param t1_mat matrix of word vectors for first set of target words
  #' @param t2_mat matrix of word vectors for second set of target words
  #' @param a1_mat matrix of word vectors for first set of attribute words
  #' @param a2_mat matrix of word vectors for second set of attribute words

  ##### Target 1 Test
  # Get effect size and p-value
  mean_t1_a1 <- calc_word_distances(t1_mat, a1_mat, "WEAT")
  mean_t1_a2 <- calc_word_distances(t1_mat, a2_mat, "WEAT")
  # get summation between targ1/att1 and targ1/att2 associations
  # (see "Methods" section for formula)
  sum_t1 <- sum(mean_t1_a1 - mean_t1_a2)

  ##### Target 2 Test
  mean_t2_a1 <- calc_word_distances(t2_mat, a1_mat, "WEAT")
  mean_t2_a2 <- calc_word_distances(t2_mat, a2_mat, "WEAT")
  # get summation between targ2/att1 and targ2/att2 associations
  sum_t2 <- sum(mean_t2_a1 - mean_t2_a2)

```

```

# Difference between summations for target1 and target2
# THIS IS THE TEST STATISTIC
t <- sum_t1 - sum_t2

# EFFECT SIZE
effect <- .calc_effect_size(mean_t1_a1, mean_t1_a2, mean_t2_a1, mean_t2_a2)

return(c(t, effect))
}

.calc_effect_size <- function(mean_t1_a1, mean_t1_a2, mean_t2_a1, mean_t2_a2) {
  #' Returns the effect size, i.e. normalized measure of how separated the two
  #' distributions are
  #' @param mean_t1_a1 avg cosine distances for each word in first vector of target words
  #' against first vector of attribute words
  #' @param mean_t1_a2 avg cosine distances for each word in first vector of target words
  #' against second vector of attribute words
  #' @param mean_t2_a1 avg cosine distances for each word in second vector of target words
  #' against first vector of attribute words
  #' @param mean_t2_a2 avg cosine distances for each word in second vector of target words
  #' against second vector of attribute words

  #calculate effect size (see "Methods" section for formula)
  mean_t1 <- mean(mean_t1_a1 - mean_t1_a2)
  mean_t2 <- mean(mean_t2_a1 - mean_t2_a2)

  diff_tot <- c(mean_t1_a1 - mean_t1_a2, mean_t2_a1 - mean_t2_a2)

  std <- sd(diff_tot)

  #EFFECT SIZE
  effect <- abs((mean_t1 - mean_t2)/std)

  return(effect)
}

```