Bayesian Explanations of Harvard: College Scorecard Dataset

Michael L. Thompson

October 15, 2018

Contents

Introduction

This **R** Markdown document applies the *Generalized Bayes Factor (GBF)* to find the *Most Relevant Explanation* as described by Yuan, Lim, and Lu (2011). The context in which I apply GBF is the U.S. Department of Education College Scorecard dataset, focusing specifically on **Harvard University**.

Using the dataset trimmed down to ~1100 4-year public and private non-profit universities, I estimate the conditional probability distributions of student population factors such as ethnicity, SAT score, and discipline studied. The resultant model of the dataset is a *Tree-Augmented Naive Bayes Net.* I then apply this model to answer questions such as "Which colleges are most representative of a given student profile?" and "Which student profiles are most representative of a given college?"

Generalized Bayes Factor

The generalized Bayes factor $\mathrm{GBF}(h;e)$ quantifies the degree to which an hypothesis $H = h$ *explains* or is *representative* of specified evidence $E = e$. The formula is given in terms of conditional probabilities as follows:

$$\begin{aligned}\mathrm{GBF}(h;e) &\equiv \frac{\mathrm{P}(E=e|H=h)}{\mathrm{P}(E=e|H\neq h)}\\ &= \frac{\mathrm{Odds}(H=h|E=e)}{\mathrm{Odds}(H=h)}\end{aligned}$$

where the *prior odds in favor of H=h* are $\mathrm{Odds}(H=h) \equiv \frac{\mathrm{P}(H=h)}{\mathrm{P}(H\neq h)}$, and analogously, the *posterior odds in favor of H=h given evidence E=e* are $\mathrm{Odds}(H=h|E=e) \equiv \frac{\mathrm{P}(H=h|E=e)}{\mathrm{P}(H\neq h|E=e)}$.

In the cases of simple hypotheses on parameter values about data modeled statistically as evidence, these conditional probabilities correspond to *likelihoods.* But, as I.J. "Jack" Good explains in Good (1985), $\mathrm{GBF}(h;e)$ is applicable to arbitrarily complex hypotheses and evidence sets and the probabilities are more flexibly defined in terms of Bayesian subjective probabilities. Moreover, Good (1985) goes on to show that WE(h;e) $\equiv$ $\log(\mathrm{GBF}(h;e))$ is the *Weight of Evidence* in favor of hypothesis $H = h$; and WE(h;e) has connections to the *Kullback-Leibler Divergence*, KLD, from Information Theory (also known as Communications Theory).

See also Tenenbaum and Griffiths (2001), Fitelson (2007) and Pacer et al. (2013) for discussions of why and how well $\mathrm{GBF}(h;e)$ measures the explanatory or representative strength of h with respect to e, especially relative to human judgments.

Application to Colleges and Student Profiles

Note that in answering "Which colleges are most representative of a given student profile?", the given evidence $E = e = \{\text{student traits}\}$ is the student profile; and the candidate hypotheses $H = h_i = \{\text{college } i\}$ are drawn from the list of colleges for which we have data.

In such a scenario, $\mathrm{GBF}(h = \text{college } i; e = \{\text{student traits}\})$ measures how much more *probable, i.e., prevalent,* students are with these traits amongst students at college i than they are amongst all students not attending that college i, which is roughly the entire student population.

Since any single college contributes negligibly to the overall student population of the more than 1100 4-year colleges considered, $\mathrm{P}(E = e|H \neq h) \approx \mathrm{P}(E = e)$, in which case $\mathrm{GBF}(h;e)$ is approximately equal to the so-called *belief update ratio*, $\mathrm{BUR}(h;e) \equiv \frac{\mathrm{P}(E=e|H=h)}{\mathrm{P}(E=e)} = \frac{\mathrm{P}(H=h|E=e)}{\mathrm{P}(H=h)}$.

(In Bayes' Theorem, $\mathrm{BUR}(h;e)$ is the multiplicative factor applied to the prior belief $\mathrm{P}(H = h)$, updating it into the posterior belief $\mathrm{P}(H = h|E = e)$, hence the name "belief update ratio".)

Conversely, in answering “Which student profiles are most representative of a given college?”, the given evidence $E = e = \{\text{college } i\}$ is a specific college, and the candidate hypotheses $H = h = \{\text{student traits}\}$ are all possible instantiations of the student traits.

In such a scenario, GBF($h = \{\text{student traits}\}; e = \text{college } i$) measures how much greater the *odds in favor* of a student randomly drawn from amongst the students at college i having the traits of the given profile are than the odds in favor of drawing such a student from amongst the entire college student population. In other words, the student profile having the greatest GBF($h = \{\text{student traits}\}; e = \text{college } i$) is the one that most distinctively sets college i apart from all the other colleges.

Findings

I find the following specifically about Harvard University:

- The student profile having the greatest GBF($h = \{\text{student traits}\}; e = \text{college } i = \text{"Harvard University"}$), thus being the one that most distinctively sets Harvard apart from all the other colleges, is $h = \{SAT > 1400\}$.
  - GBF($h = \{SAT > 1400\}; e = \text{"Harvard University"}) = 292$
  - So, the *odds in favor* of a student randomly drawn from amongst Harvard students having SAT > 1400 are ≈ 292× greater than the *odds in favor* of randomly drawing such a student from amongst the entire college student population!
- When specifying this Harvard-representative profile as evidence $e = \{SAT > 1400\}$ and ordering colleges as candidate hypotheses $h = \text{college } i$ from largest GBF to smallest, the top 10 colleges, with their GBF($h = \textit{college}; e = \{SAT > 1400\}$) values, are the following:
  1. California.Institute.of.Technology 28.5
  2. University.of.Chicago 27.5
  3. Massachusetts.Institute.of.Technology 27.2
  4. Yale.University 27.2
  5. Princeton.University 27.1
  6. Harvard.University 27.0
  7. Washington.University.in.St.Louis 26.2
  8. Harvey.Mudd.College 25.6
  9. Vanderbilt.University 25.0
  10. Stanford.University 24.8
  - In other words, the expected colleges have nearly the same large incidence of high-SAT students distinguishing those colleges from other American colleges. And, we see that students with SAT > 1400 are 27× more prevalent at Harvard than they are amongst the general college student population.
- Harvard University has as its most representative student profile, of all those profiles involving an ethnicity amongst U.S. students, the student profile $h = \{\textit{ethnicity} = \text{Asian}, \textit{discipline} = \text{Social Sciences}\}$.
  - GBF($h = \{\textit{ethnicity} = \text{Asian}, \textit{discipline} = \text{Social Sciences}\}; e = \text{"Harvard University"}) = 6.3$
  - So, the *odds in favor* of a student randomly drawn from amongst Harvard students having traits

fitting these criteria are $\approx 6\times$ greater than the *odds in favor* of randomly drawing such a student from amongst the entire college student population.
    - Note that the next highest hypothesis is $\text{GBF}(h = \{ethnicity = \text{Asian}, discipline = \text{Science \& Engineering}\}; e = \text{"Harvard University"}) = 4.4$.
    - Also, when considering non-American ethnicities, $\text{GBF}(h = \{ethnicity = \text{Foreign}, discipline = \text{Social Sciences}\}; e = \text{"Harvard University"}) = 6.5$, which is basically the same as that above for $ethnicity = \text{Asian}$.
- When specifying this Harvard-representative profile along with SAT>1400 as evidence $e = \{SAT > 1400, ethnicity = \text{Asian}, discipline = \text{Social Sciences}\}$ and ordering colleges as candidate hypotheses $h =$ college i from largest GBF to smallest, Harvard jumps to a veritable tie for first with the University of Chicago. The top 10 colleges with their GBF values are the following:
    1. University.of.Chicago 53.4

    2. Harvard.University 52.2

    3. Princeton.University 45.3

    4. Yale.University 40.3

    5. Dartmouth.College 35.6

    6. Wellesley.College 32.5

    7. Columbia.University.in.the.City.of.New.York 32.4

    8. Duke.University 32.4

    9. Washington.University.in.St.Louis 30.5

    10. Stanford.University 27.6

    - In other words, students satisfying the evidence criteria of $\{SAT > 1400, ethnicity = \text{Asian}, discipline = \text{Social Sciences}\}$ are 52× more prevalent at Harvard than they are amongst the general college student population.
- When looking at the *least* representative profile for Harvard (i.e., $\text{GBF}(h; e = \text{"Harvard University"})$ values less than 1), if we exclude hypotheses involving low SAT scores and discipline areas for which Harvard issues few degrees, then student profile $h = \{ethnicity = \text{Black}\}$ is the sole hypothesis, having a $\text{GBF}(h = e)$ of just 0.5.
    - So, the *odds against* a randomly selected Harvard student having ethnicity $=$ Black is about $1/0.5 = 2\times$ greater than the odds against randomly selecting such a student from the general college student population.

Bar charts illustrating these findings are included in the section "Generate Explanations".

Caveats

Dated rather than Recent Data. The College Scorecard dataset used is vintage 2015, with some of the variables having been collected as early as 2005. So, it is possible that the relevant distributions of student traits have changed significantly since the collection of these data.

Aggregated rather than Student-Level Data. The student profiles I will use are defined in terms of gender, ethnicity, academic discipline, income, and SAT score. However, given that I only have aggregate data and do not have interaction information – such as the pecentages of students by ethnicity AND gender; ethnicity AND income; gender AND academic discipline; or ethnicity AND income AND SAT – the analyses presented here must be regarded as merely a demonstration of what is possible if either individual student-level data were available or aggregate percentages of 2-way (or higher order) interactions were available. To sidestep some of this issue in absence of student-level data or aggregates quantifying higher-order interactions, I'll focus only on using student profiles defined in terms of *ethnicity*, *academic discipline*, and *SAT score*.

Approximate rather than Actual Distributions. I use crude normal (Gaussian) density functions as approximations for the probability distribution of SAT scores for each college based upon the reported mean value of the combined Math and Verbal SAT scores and an assumed standard deviation of 75 points. Given finer resolution on quantiles or, better yet, student-level data, we could improve greatly upon this.

Sensitive to rather than Robust wrt Discretization/Aggregation. The results also are sensitive to the discretizations and aggregations applied to the variables in defining the student profile traits. For example, for SAT score I use 3 intervals: "A_lt1000" = SAT $\leq$ 1000, "B_lt1400" = 1000 $<$ SAT $\leq$ 1400, and "C_gt1400" = SAT $>$ 1400. Also I aggregate the academic disciplines somewhat arbitrarily into just 6 classes: "Hum" = Humanities, "SciEng" = Physical Sciences & Engineering, "SocSci" = Social Sciences, "Busnss" = Business & Management, "Tech" = Technologies, and "VisPerf" = Visual & Performing Arts. Finally, I also aggregate the ethnicities a bit arbitrarily: "white", "black", "asian", "hispanic", "foreign" = non-resident alien, and "other" = all other students. It would require greater domain knowledge and more in-depth study to identify the most appropriate discretizations/aggregations to define the student profile traits.

In sum, the analysis presented here only provides an indication of what one might do when given more appropriate data. So don't draw strong conclusions from these results.

Conclusion

Again, this is a crude analysis performed to hint at how such an approach might be useful when applied to a more complete database, namely student-level data from a large set of 4-year universities and colleges.

In light of the current lawsuit brought against Harvard University claiming discrimination against Asian-American students, it would seem from this analysis that, whether Harvard is or is not applying discriminatory practices, Asian-American students are still the most distinctive ethnic group setting Harvard's undergraduates apart from those of other colleges – and by a wide margin over even many of the comparable elite American colleges.

As a side note, the generalized Bayes factors – actually in log form, so really the weight of evidence metrics WE$(h; e)$ – are used as the feature covariates in my Kaggle scripts "Which College is Best for You_" and "Which College is Best for You, Part 2" and also in my web-app "Best Colleges for You".

Feedback

Feel free to send me feedback through LinkedIn.

-Michael L. Thompson

Technical Analysis

Below is the **R** code used to compute the GBF$(h = \{\text{student traits}\}; e = \text{college } i)$ and GBF$(h = \text{college } i; e = \{\text{student traits}\})$. Bayesian belief networks (BBN), as implemented by package `gRain`, were used to perform all of the probability calculations. The code is crude and applies brute-force enumeration to score the candidate hypotheses exhaustively. So some scenarios can take a long time to run or are just not feasible. It is recommended to use more efficient implementations of *Most Relevant Explanation, MRE* obtainable directly from Prof. Yuan of City University of New York or to do the calculations using the *SMILE/Genie* software from BayesFusion.

The use of a Tree-Augmented Naive Bayes (TAN) model, which is a specific type of Bayesian belief network (BBN) imposing strict conditional independence assumptions relative to a general BBN but less restrictive than for a Naive Bayes classifier model, allows us to summarize a large amount of data covering a high-dimensional into a concise and computationally tractable model, which is easy to interpret and efficient with which to perform Bayesian inference. However, the model shown here is the most trivial of TAN, in that only a single arc – from ethnicity to completion rate (`cmpltn`) – extends the model beyond being a simple Naive Bayes Net.

Define Functions

The main function for find the "Most Relevant ExplanationS" is `gbf_all_hypcombos` defined here. It is as brute force as it gets in performing many redundant calculations in exhaustively enumerating the candidated hypothesis variables' state combos. So it is slow. Be sure not to call it with a value of argument `n_max` greater than 4 or 5.

```
gbf_all_hypcombos <- function(bbn,ev_list, hyp_nms,phi_list,n_max=3L, min_thresh=1e-05,verbose=TRUE){

  # Make a list of lists of data.frames, each holding all of the possible combos of
  # evidence instantiation amongst up to n_max of the hyp_nms, grabbing state
  # levels from phi_list.
  hyp_list <- seq_len(pmin(n_max,length(hyp_nms))) %>%
    map(
      ~ combn(length(hyp_nms),.x) %>%
        apply(2,function(i) expand.grid(phi_list[hyp_nms[i]],stringsAsFactors = TRUE ))
    )

  gbf_df <- hyp_list %>% map_dfr(
    ~ bind_rows(.x) %>%
      rowwise() %>%
      do(
        {
          hyp     <- .[!sapply(.,is.na)]
          hyp_int <- imap_int(hyp,~ which(bbn$universe$levels[[.y]]==.x)) #sapply(hyp,as.integer)

          # Prior Probability of H
          p_x   <- bbn %>% querygrain( nodes=names(hyp), type='joint') %>%
            {.[matrix(hyp_int[names(dimnames(.))],nrow=1)]} #%T>% print()
          #Posterior Probability of H given E
          # print(
          #   list(
          #     index=hyp_int,
          #     value=hyp,
          #     dimension=names(dimnames(querygrain(bbn, nodes=names(hyp),evidence=ev_list, type='joint
          #   ) %>% c(list(new_index=.$index[.$dimension]))
          # )
          p_x_e <- bbn %>% querygrain( nodes=names(hyp),evidence=ev_list, type='joint' ) %>%
          {.[matrix(hyp_int[names(dimnames(.))],nrow=1)]}

          o_x   <- p_x/(1-p_x) # Prior Odds in favor of H
          o_x_e <- p_x_e/(1-p_x_e) # Posterior Odds in favor of H given E
          gbf   <- pmax(min_thresh,o_x_e/o_x) # Generalized Bayes Factor, GBF(H|E)
          bur   <- p_x_e/p_x # Belief Update Ratio
          bind_cols(
            data_frame(
              p_x=p_x,p_x_e=p_x_e,bur=bur,o_x=o_x,o_x_e=o_x_e,gbf=gbf,
              terms=paste(sprintf("%s=%s",names(hyp),sapply(hyp,as.character)),collapse=',')
            ),
            as_tibble(.)
          )
        }
      ) %>%
      ungroup()
  ) %>%
    arrange(desc(gbf),nchar(terms))

  # Define the thresholds for interpretation of the Bayes factors,
  # see https://en.wikipedia.org/wiki/Bayes_factor#Interpretation
  gbf_threshold <- 10^c(
    'neither'    = 0,
    'not worth mentioning'=0.25,
    'barely\nworth mentioning'=0.5,
    'substantial'= 1,
    'strong'     = 1.5,
    'very strong'= 2,
    'decisively' = Inf
  )
  get_ithresh <- function(x){ pmin(length(gbf_threshold),length(which(x>=gbf_threshold))+1) }

  # Create a data frame that provides an interpretive 'support' phrase for each hypothesis
  terms     <- gbf_df$terms %>% setNames(.,.)
  terms_rev <- gbf_df %>% arrange(gbf,nchar(terms)) %$% terms %>% setNames(.,.)
  gbf_df %<>%
    mutate(
      gbf = gbf + 1e-15,
      hypothesis = factor(terms,terms),
      support = ifelse(
        gbf>=1,
        map_chr(gbf,  ~ sprintf("supports: %s", names(gbf_threshold)[get_ithresh(.x)])),
        map_chr(1/gbf,~ sprintf("refutes:  %s", names(gbf_threshold)[get_ithresh(.x)]))
      ),
      support = factor(
        support,
        levels=rev(c(
          sprintf("refutes:  %s",rev(names(gbf_threshold[-1])) ),
          sprintf("supports: %s", names(gbf_threshold)))
        )
      )
    ) %>%
    dplyr::select(-terms)  %>%
    dplyr::select(hypothesis,gbf,support,everything())

  # Strip the explanations down to the Most Relevant Explanation
  # (i.e., the minimal explanation, which is concise and not dominated
  # by a more concise explanation).
  # An explanation (hypothesis) is dominated if there is a simpler hypothesis
  # (one with a subset of its clauses) that is more strongly supported/refuted
```

```
  # by the evidence than the explanation is supported/refuted by the evidence.
  dominated <- seq_along(terms) %>%
    map_lgl(
      ~ {
        i <- .x;
        tmp<-str_split(terms[i],",")[[1]];
        any( map_lgl(str_split(terms[1:i][-i],","), ~ length(setdiff(.x,intersect(tmp,.x)))==0))
      }
    )
  #rterms <- rev(terms)
  dominated_neg <- seq_along(terms_rev) %>% setNames(names(terms_rev)) %>%
    map_lgl(
      ~ {
        i <- .x;
        tmp <- str_split(terms_rev[i],",")[[1]];
        any( map_lgl(str_split(terms_rev[1:i][-i],","), ~ length(setdiff(.x,intersect(tmp,.x)))==0))
      }
    ) %>%
    {.[names(terms)]} # make sure aligned in same order as dominated.
  gbf_min_df <- gbf_df %>%
    filter(!ifelse(gbf>=1,dominated,dominated_neg)) %>%
    filter(!grepl("not worth",support)) %T>%
    {if(verbose){print()}}

  return( list(ev_list = ev_list, hyp_nms = hyp_nms, gbf = gbf_df, gbf_min = gbf_min_df ) )
}
```

Prepare the Data

I derived the data_frame we'll use from the U.S. Department of Education, College Scorecard database as downloaded from Kaggle.com. Here, I'll just load in a subsetted (by rows and columns) version of the dataset that spans ~1120 4-year public and private non-profit colleges.

```
# Load the pre-processed College Scorecard dataset.  Object `DataSpec` is built
# using script "buildStaticDB.R", which is part of the "Best Colleges for You"
# app by M.L. Thompson, copyright 2016, and is made available under the Apache
# License 2.0.
load( "college_data.RData" , verbose = TRUE )
```

```
## Loading objects:
##   college_data
```

```
#college_data %>% print()

# Names of all of the discipline classes in the College Scorecard dataset.
dnm <- c(
  "AgricultureAgriculture" , "NaturalResources","ArchitectureRelated",
  "AreaEthnic","CommunicationJournalism" ,
  "CommunicationsTechnologies", "ComputerInformation","PersonalCulinary",
  "Education", "Engineering" , "EngineeringTechnologies" ,
  "ForeignLanguages", "FamilyConsumer" , "LegalProfessions" ,
  "EnglishLanguage", "LiberalArts" , "LibraryScience" ,
  "BiologicalBiomedical" , "MathematicsStatistics" ,
  "MilitaryTechnologies" , "MultiInterdisciplinary" , "ParksRecreation" ,
  "PhilosophyReligious", "TheologyReligious","PhysicalSciences" ,
  "ScienceTechnologies" , "Psychology" ,
  "HomelandSecurity" , "PublicAdministration" , "SocialSciences",
  "ConstructionTrades" , "MechanicRepair",
  "PrecisionProduction" , "TransportationMaterials" , "VisualPerforming",
  "HealthProfessions", "BusinessManagement" ,
  "History"
)

# Abbreviated names for aggregate clusters of discipline classes. (must be exact
# same length as vector `dnm`)
dnm_agg <- c("Tech","Tech","SciEng","Hum","Hum","Tech","SciEng","Tech",
             "Hum","SciEng","Tech","Hum","Tech","Hum","Hum","Hum",
             "Tech","SciEng","SciEng","Tech","Hum","Tech","Hum","Hum",
             "SciEng","Tech","SocSci","Tech","Tech","SocSci","Tech",
             "Tech","Tech","Tech","VisPerf","Tech","Busnss","Hum")

# Build the smaller data_table to be used as the basis of the Tree-Augmented
# Naive Bayes (TAN) network model of some select attributes of each college and
# the student body of students at each college. Each row of the
# `DataSpec$student` data_table corresponds to a college. The resulting object
# `adf` is a list of lists & vectors to be used to create all of the conditional
# probability tables (CPT) needed to define the structure and parameters of the
# TAN model.
adf <- college_data %>%
  {
    # capture the raw ingredients for conditional tables of key factors into a
    # list of lists, each top-level element named by a school, and each
    # bottom-level element representing the level of a factor.
    setNames(
      apply(
        .,
        1,
        function(x) {
          x <- lapply(x[-1],as.numeric);
          list(
            # SAT (quatile intervals from approximated distribution assuming all
            # colleges have same standard dev. of 75 pts.)
            sat = pnorm(c(1000,1400),x$SAT_AVG,75) %>%
              {list(A_lt1000= .[1],B_lt1400= .[2] - .[1],C_gt1400=1 - .[2])},
            # DISCIPLINE (lumped into discipline clusters named by vector `dnm_agg`)
            disc = summarize(
              group_by(data_frame(dnm=as.numeric(x[dnm]),agg=dnm_agg),agg),
              disc=sum(dnm)
              ) %$%
              as.list(setNames(disc,agg)),
            # GENDER (binary, proportions "female" and "not_female")
            gender = list(
              female     = x$female_2005,
              not_female = 1.0 - x$female_2005
            ),
            # INCOME, annual, (ternary, labeled "A_lt30K","B_lt110K", and "C_gt110K")
            income  = list(
              A_lt30K  = x$DEP_INC_PCT_LO,
              B_lt110K = 1 - (x$DEP_INC_PCT_LO+x$DEP_INC_PCT_H2),
              C_gt110K = x$DEP_INC_PCT_H2
            ),
            # ETHNICITY (lumping all other than white, black, asian, and
            # hispanic into "other", which include non-resident aliens)
            ethnicity = list(
              white    = x$UGDS_WHITE,
              black    = x$UGDS_BLACK,
              asian    = x$UGDS_ASIAN,
              hispanic = x$UGDS_HISP,
              foreign  = x$UGDS_NRA,
              other    = sum(
                as.numeric(x[c('UGDS_AIAN','UGDS_NHPI','UGDS_2MOR','UGDS_UNKN')]),
                na.rm=TRUE
              )
            ),
            # COMPLETION (binary by ethnicity, labeled "yes" and "no"
            # correponding to "completed" and "not completed")
            # This is the sole TAN edge that extends model beyond simple Naive Bayes net.
            cmpltn   =  c(
              x[c("C150_4_WHITE" ,"C150_4_BLACK","C150_4_ASIAN","C150_4_HISP","C150_4_NRA")],
              list(C150_other=x$C150_4_POOLED_SUPP)
            ) %>% lapply(function(x) ifelse(is.na(x),0,x)),
            # LOAN DEFAULT RATES (binary, labeled "yes" & "no" corresponding to
            # "default" and "not default")
            loandflt = list(yes = x$CDR3, no = 1 - x$CDR3),
            # EARNINGS 6YRS AFTER ENTRY (binary, <=$50,000/yr and >$50,000/yr)
            earnings = pnorm(50000,x$mn_earn_wne_p6_2005,x$sd_earn_wne_p6_2005) %>%
              {list(A_lteq50K= .[1],B_gt50K = 1 - .[1])},
            # ADMISSION RATES (binary, labeled "yes" & "no" corresponding to
            # "admitted" and "not admitted")
            admssn  = list(yes = x$ADM_RATE, no = 1 - x$ADM_RATE)          )
        }
      ),
      .$id
    )
  }

adf <- adf[-214] # for some reason the 214th one messes up

ethnm   <- adf[[1]]$ethnicity %>% names()
coll_nm <- gsub("\\.+",".",make.names(names(adf)))

gg_color_hue <- function(n) {
  hues  = seq(15, 375, length = n + 1)
  hcl(h = hues, l = 65, c = 100)[1:n]
}

gbf_threshold <- 10^c(
  'neither'    = 0,
  'not worth mentioning'=0.25,
  'barely\nworth mentioning'=0.5,
  'substantial'= 1,
  'strong'     = 1.5,
  'very strong'= 2,
  'decisively' = Inf
)
gg_colors <- gbf_threshold %>%
  {setNames(gg_color_hue(length(.)),sprintf("supports: %s",names(.)))}
```

Build a Model: TAN BBN

Build the tree-augmented naive Bayes (TAN) Bayesian Belief Network (BBN) model.

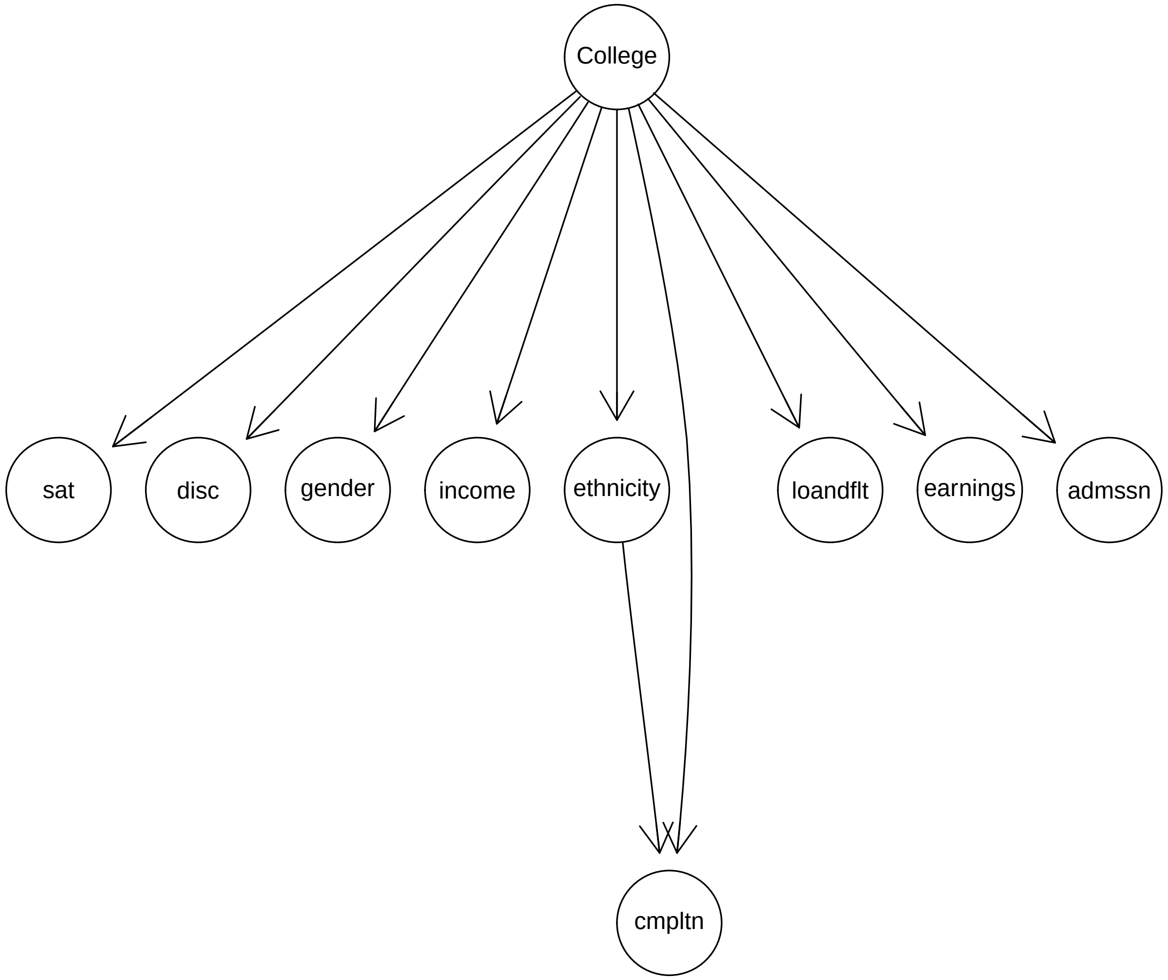
```
# Convert the list of lists/vectors of probabilities from DataSpec$student into
# a list of conditional probability tables.
cptlist <- adf[[1]] %>%
  names() %>%
  setNames(.,.) %>%
  map( ~ map_dfr(adf,.x) ) %>%
  imap( function(.x,.y){
    if(.y != "cmpltn"){
      cptable(
        as.formula(sprintf("~ %s + College",.y)),
        values = c(t(set_rownames(as.matrix(.x),names(adf)))) %>% {ifelse(is.na(.),0,.)},
        levels = colnames(.x)
      )
    } else {
      cptable(
        as.formula(sprintf("~ %s + College + ethnicity",.y)),
        values = colnames(.x) %>%
          sapply(
            function(ethn) c(t(set_rownames(as.matrix(.x),names(adf)))[ethn,]) %>%
              {ifelse(is.na(.),0,.)} %>%
              rbind(1- . )
          ),
        levels = c("yes","no")
      )
    }
  }
  ) %>%
  c(
    list(
      College = cptable(
        "College",
        values= college_data$UGDS[1:length(adf)],
        levels=gsub("\\.+",".",make.names(names(adf)))
      )
    )
  )

# Make the TAN Bayesian belief network (BBN) model.
bbn <- cptlist %>%
  compileCPT() %>%
  grain(smooth=1.0e-4)
```

Show the marginal probabilities of all the factors, as captured by the TAN BBN.

```
# Query the model (performing Bayesian inference) to generate marginal distributions.
bbn %>%
  querygrain(nodes= names(cptlist)[-length(cptlist)]) %>%
  {print(list(`Marginal Distributions (All)`=.))}
```

```
## $`Marginal Distributions (All)`
## $`Marginal Distributions (All)`$sat
## sat
##   A_lt1000   B_lt1400   C_gt1400
## 0.30568962 0.66040524 0.03390514
##
## $`Marginal Distributions (All)`$disc
## disc
##     Busnss        Hum     SciEng     SocSci       Tech    VisPerf
## 0.18077166 0.24145326 0.16197102 0.14730552 0.20904298 0.05945557
##
## $`Marginal Distributions (All)`$gender
## gender
##     female not_female
##  0.5800994  0.4199006
##
## $`Marginal Distributions (All)`$income
## income
##   A_lt30K  B_lt110K  C_gt110K
## 0.2739213 0.6260295 0.1000492
##
## $`Marginal Distributions (All)`$ethnicity
## ethnicity
##      white      black      asian   hispanic    foreign      other
## 0.56946930 0.12451767 0.06519361 0.11966424 0.04221759 0.07893759
##
## $`Marginal Distributions (All)`$cmpltn
## cmpltn
##       yes        no
## 0.5752357 0.4247643
##
## $`Marginal Distributions (All)`$loandflt
## loandflt
##        yes         no
## 0.07255725 0.92744275
##
## $`Marginal Distributions (All)`$earnings
## earnings
## A_lteq50K   B_gt50K
## 0.6585532 0.3414468
##
## $`Marginal Distributions (All)`$admssn
## admssn
##       yes        no
## 0.6113386 0.3886614
```

```
# Show the model structure.
bbn %>% plot()
```



Query the Bayesian Model

Let’s see the conditional probability of `ethnicity` at `College="Harvard University"`:

$$\mathrm{P}(ethnicity|College = \text{"Harvard University"})$$

.

WARNING: function `gRain::querygrain()` does not report an error when the evidence is incorrect or nonsensical. It will just return the population marginal distributions! Also, all node names and evidence states are case-sensitive. To avoid typos & mis-specification that would generate erroneous results – without the benefit of an error message – use regular expression searches (e.g. `grep()`) to retrieve node state level values from the BBN’s `universe` object, which defines all nodes and states used in the model.

```
bbn %>%
  querygrain(nodes="ethnicity",type="joint") %>%
  multiply_by(100) %>%
  round(1) %>%
  {print(list(`Distribution (All)`=.))}
```

```
## $`Distribution (All)`
## ethnicity
##    white    black    asian hispanic  foreign    other
##     56.9     12.5      6.5     12.0      4.2      7.9
```

```
# REMEMBER: College names are the same as those in character vector `coll_nm`
# and have all spaces replaced with periods!
bbn %>%
  querygrain(
    nodes    = "ethnicity",
    evidence = c(College = grep("Harvard.+Univ",bbn$universe$levels$College,value=TRUE)),
    type     = 'joint'
  ) %>%
  multiply_by(100) %>%
  round(1) %>%
  {print(list(`Distribution (Harvard)`=.))}
```

```
## $`Distribution (Harvard)`
## ethnicity
##    white    black    asian hispanic  foreign    other
##     46.2      6.5     17.8      9.3     10.3      9.9
```

Generate Explanations

WARNING: The function `gbf_all_hypcombos()` calls function `gRain::querygrain()`, which, again, does not report an error when the evidence list is incorrect or nonsensical. It will just return the population marginal distributions! Also, all node names and evidence states are case-sensitive. Note that node `"College"` is the only one with a capitalized name, and it **must** be capitalized.

Most Representative Student Profiles for Harvard

In the table for `gbf_harvard$gbf_min` that is output, the `"x"` in column names `p_x`, `p_x_e`, `o_x`, and `o_x_e` refers to the hypothesis $H = h$. (Names coded this way are a hold over from an earlier project).

```
# Evidence to assert: in this case, it's the single proposition "Student from Harvard"
ev_list <- list(
  College = grep(
    'Harvard.+University',
    bbn$universe$levels$College,
    value=TRUE
  )
) %T>% {print(list(Evidence=.))}
```

```
## $Evidence
## $Evidence$College
## [1] "X166027_Harvard.University"
```

```
# Candidate variables to explore as hypotheses explaining the evidence
hyp_nms <- c("ethnicity","disc","sat") %T>%
  {print(list(`Hypothesis Space`=bbn$universe$levels[.]))}
```

```
## $`Hypothesis Space`
## $`Hypothesis Space`$ethnicity
## [1] "white"    "black"    "asian"    "hispanic" "foreign"  "other"
##
## $`Hypothesis Space`$disc
## [1] "Busnss"  "Hum"     "SciEng"  "SocSci"  "Tech"    "VisPerf"
##
## $`Hypothesis Space`$sat
## [1] "A_lt1000" "B_lt1400" "C_gt1400"
```

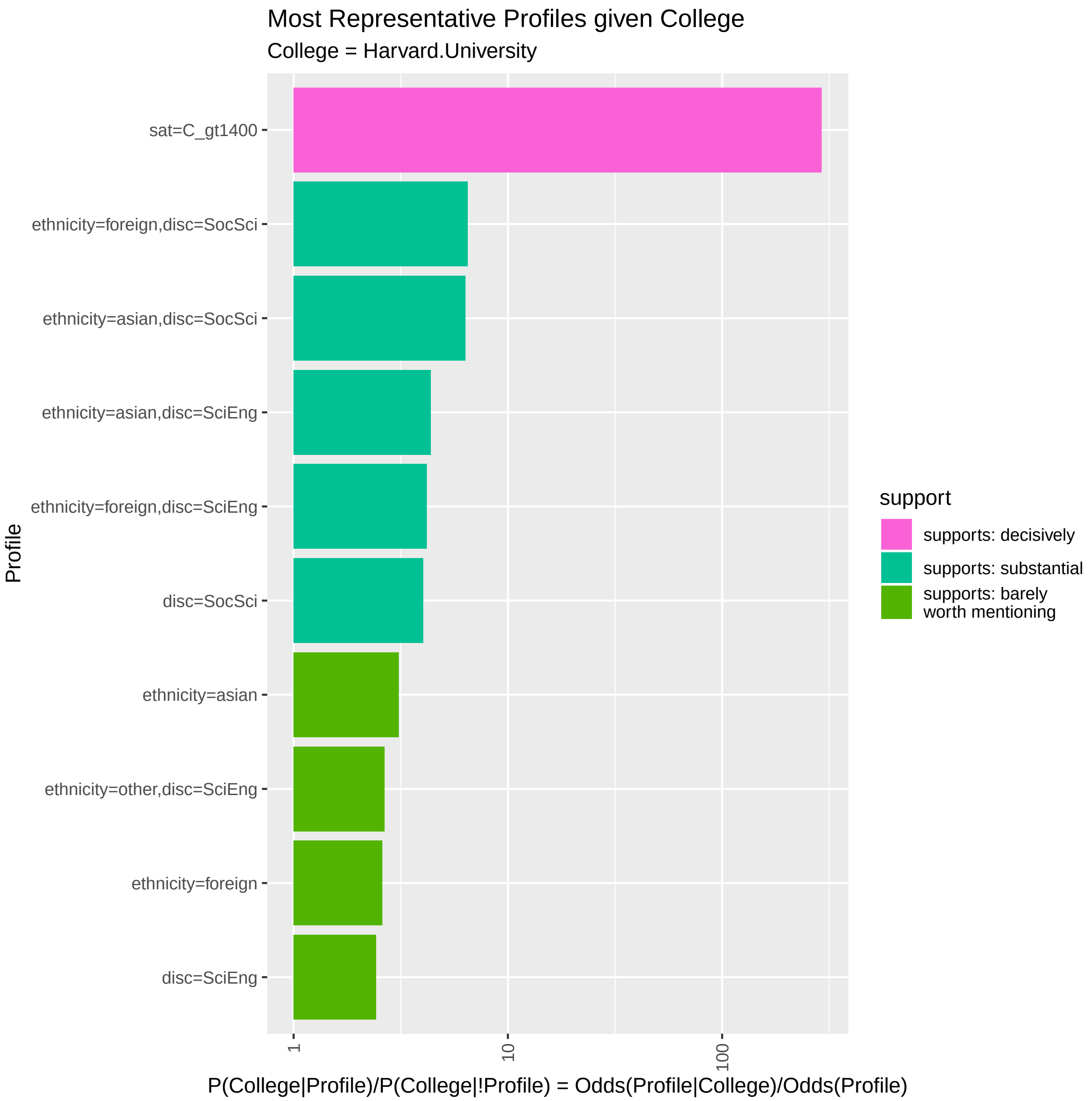
```
gbf_harvard <- bbn %>%
  gbf_all_hypcombos(
    ev_list  = ev_list,
    hyp_nms  = hyp_nms,
    phi_list = .$universe$levels[hyp_nms],
    n_max    = 4L,
    verbose  = FALSE
  )
```

```
gbf_harvard$gbf_min %>%
  dplyr::select(-matches("^(o|p)_|bur")) %>%
  {print(list("Minimal Explanation"= .))}
```

```
## $`Minimal Explanation`
## # A tibble: 29 x 6
##    hypothesis             gbf support               ethnicity disc  sat
##    <fct>                <dbl> <fct>                 <fct>     <fct> <fct>
##  1 sat=C_gt1400        292.   supports: decisively  <NA>      <NA>  C_gt1~
##  2 ethnicity=foreign,~   6.48 supports: substantial foreign   SocS~ <NA>
##  3 ethnicity=asian,di~   6.34 supports: substantial asian     SocS~ <NA>
##  4 ethnicity=asian,di~   4.36 supports: substantial asian     SciE~ <NA>
##  5 ethnicity=foreign,~   4.19 supports: substantial foreign   SciE~ <NA>
##  6 disc=SocSci           4.03 supports: substantial <NA>      SocS~ <NA>
##  7 ethnicity=asian       3.10 "supports: barely\nw~ asian     <NA>  <NA>
##  8 ethnicity=other,di~   2.66 "supports: barely\nw~ other     SciE~ <NA>
##  9 ethnicity=foreign     2.60 "supports: barely\nw~ foreign   <NA>  <NA>
## 10 disc=SciEng           2.42 "supports: barely\nw~ <NA>      SciE~ <NA>
## # ... with 19 more rows
```

Let's plot the analysis results from the above table of most representative profiles for Harvard:

```
ev_str <- gsub("^.+_","",gbf_harvard$ev_list[[1]])
gbf_harvard$gbf_min %>%
  filter(gbf > 1) %>%
  mutate(Profile = as.character(hypothesis) %>% factor(.,rev(.))) %>%
  {
    ggplot(.,aes(x=Profile,y=gbf,fill=support))+
      geom_bar(stat='identity')+
      scale_y_log10() +
      scale_fill_manual(values = gg_colors) +
      theme(axis.text.x = element_text(angle=90,hjust=1,vjust=0.5))+
      ylab("P(College|Profile)/P(College|!Profile) = Odds(Profile|College)/Odds(Profile)") +
      coord_flip() +
      ggtitle(
        "Most Representative Profiles given College",
        sprintf("College = %s",ev_str)
      )
  } %>%
  print()
```

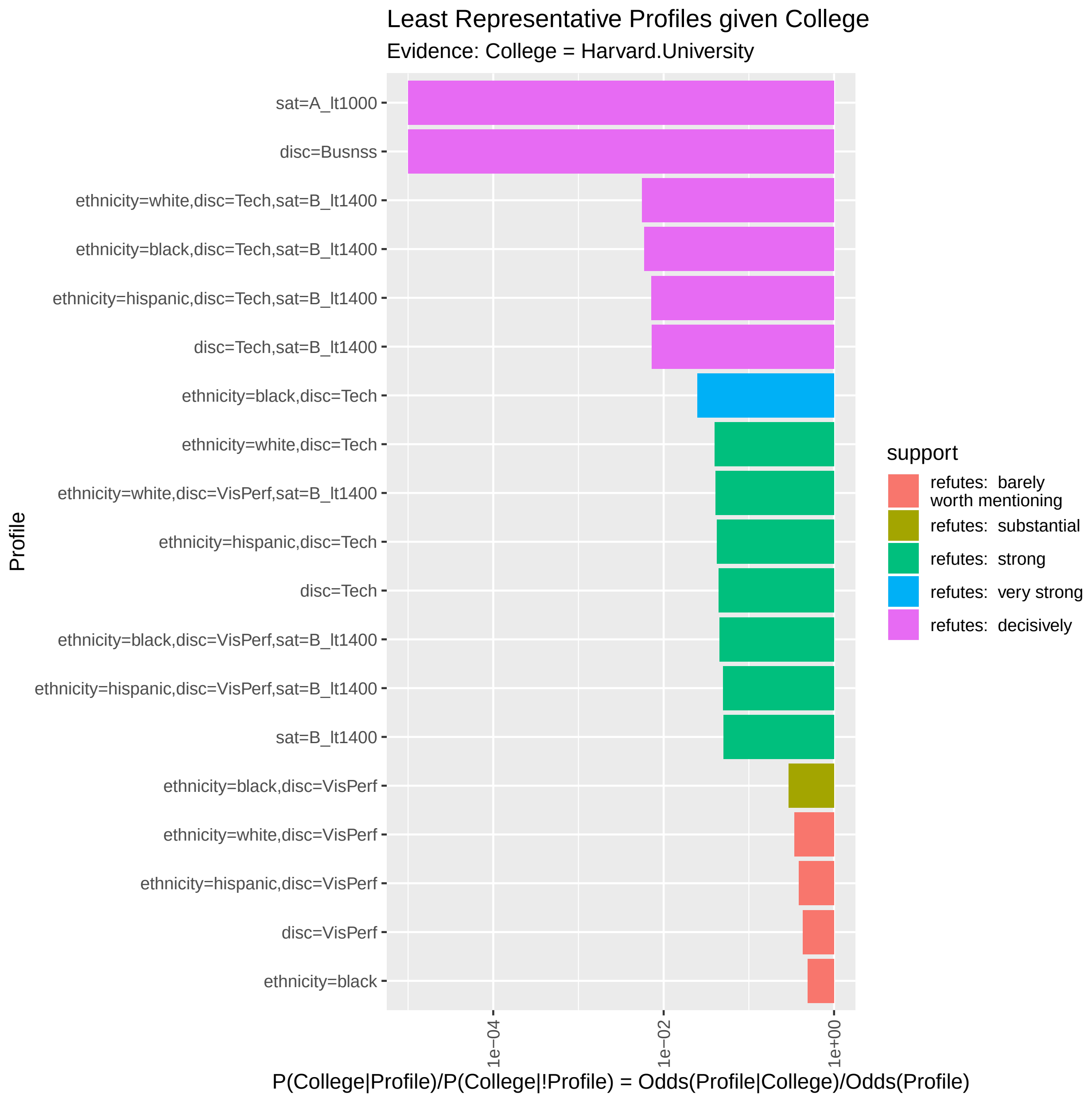


NOTE: The plots use a logarithmic scale for the x-axis, which captures GBF$(h; e)$, therefore the bars are on a scale measuring the weight of evidence in support of $H = h$. The y-axis in all the plots is always the candidate hypotheses, H, regardless of whether they are student profiles or colleges. The evidence is always listed beneath the plot's main title.

Least Representative Student Profiles for Harvard

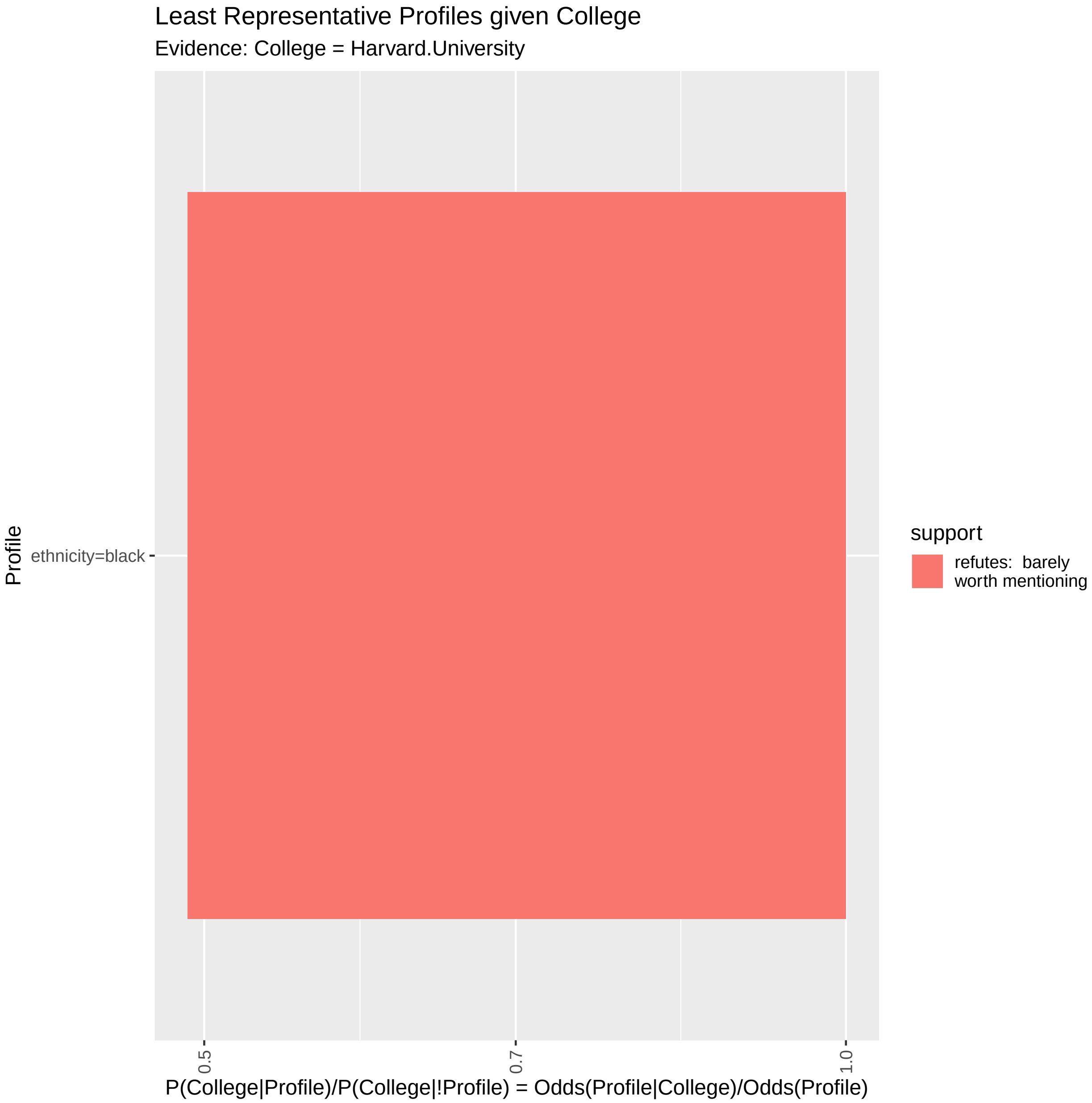
For contrast, these are the least representative profiles for Harvard:

```r
ev_str <- gsub("^.+_","",gbf_harvard$ev_list[[1]])
gbf_harvard$gbf_min %>%
  filter(gbf < 1) %>% top_n(25,wt=-gbf) %>%
  mutate(Profile = as.character(hypothesis) %>% factor(.,(.))) %>%
  {
    ggplot(.,aes(x=Profile,y=gbf,fill=support))+
      geom_bar(stat='identity')+
      scale_y_log10() +
      theme(axis.text.x = element_text(angle=90,hjust=1,vjust=0.5))+
      ylab("P(College|Profile)/P(College|!Profile) = Odds(Profile|College)/Odds(Profile)") +
      coord_flip() +
      ggtitle(
        "Least Representative Profiles given College",
        sprintf("Evidence: College = %s",ev_str)
      )
  } %>%
  print()
```



Remarkably, if we consider only high SAT and disciplines in {“Hum”,“SciEng”,“SocSci”}, then the only “Least Representative” student profile is $\{ethnicity =$ Black $\}$. Albeit, only mildly so. This is consistent with an article from Nov. 23, 2015 in *The Atlantic* “The Missing Black Students at Elite American Universities” by Andrew McGill.

```r
ev_str <- gsub("^.+_","",gbf_harvard$ev_list[[1]])
gbf_harvard$gbf_min %>%
  filter(gbf < 1) %>%
  # drop low sat hypotheses and small disciplines hypotheses
  filter(!(sat %in% c("A_lt1000","B_lt1400")), !(disc %in% c("Tech","VisPerf","Busnss"))) %>%
  top_n(25,wt=-gbf) %>%
  mutate(Profile = as.character(hypothesis) %>% factor(.,(.))) %>%
  {
    ggplot(.,aes(x=Profile,y=gbf,fill=support))+
      geom_bar(stat='identity')+
      scale_y_log10() +
      theme(axis.text.x = element_text(angle=90,hjust=1,vjust=0.5))+
      ylab("P(College|Profile)/P(College|!Profile) = Odds(Profile|College)/Odds(Profile)") +
      coord_flip() +
      ggtitle(
        "Least Representative Profiles given College",
        sprintf("Evidence: College = %s",ev_str)
      )
  } %>%
  print()
```



Most Representative Colleges

Only Considering High SAT

Let's see what other colleges are representative of the top student profile from Harvard as evidence:

$$\text{GBF}(h = \text{college } i; e = \{SAT > 1400\})$$

.

We see that Harvard comes in 6th, and the ordering of colleges deviates only slightly from a sorting by highest average SAT.

```
ev_list  <- list(sat="C_gt1400")
hyp_nms  <- "College"
gbf_prof <- bbn %>%
  gbf_all_hypcombos(
    ev_list  = ev_list,
    hyp_nms  = hyp_nms,
    phi_list = .$universe$levels[hyp_nms],
    verbose  = FALSE
  )
```

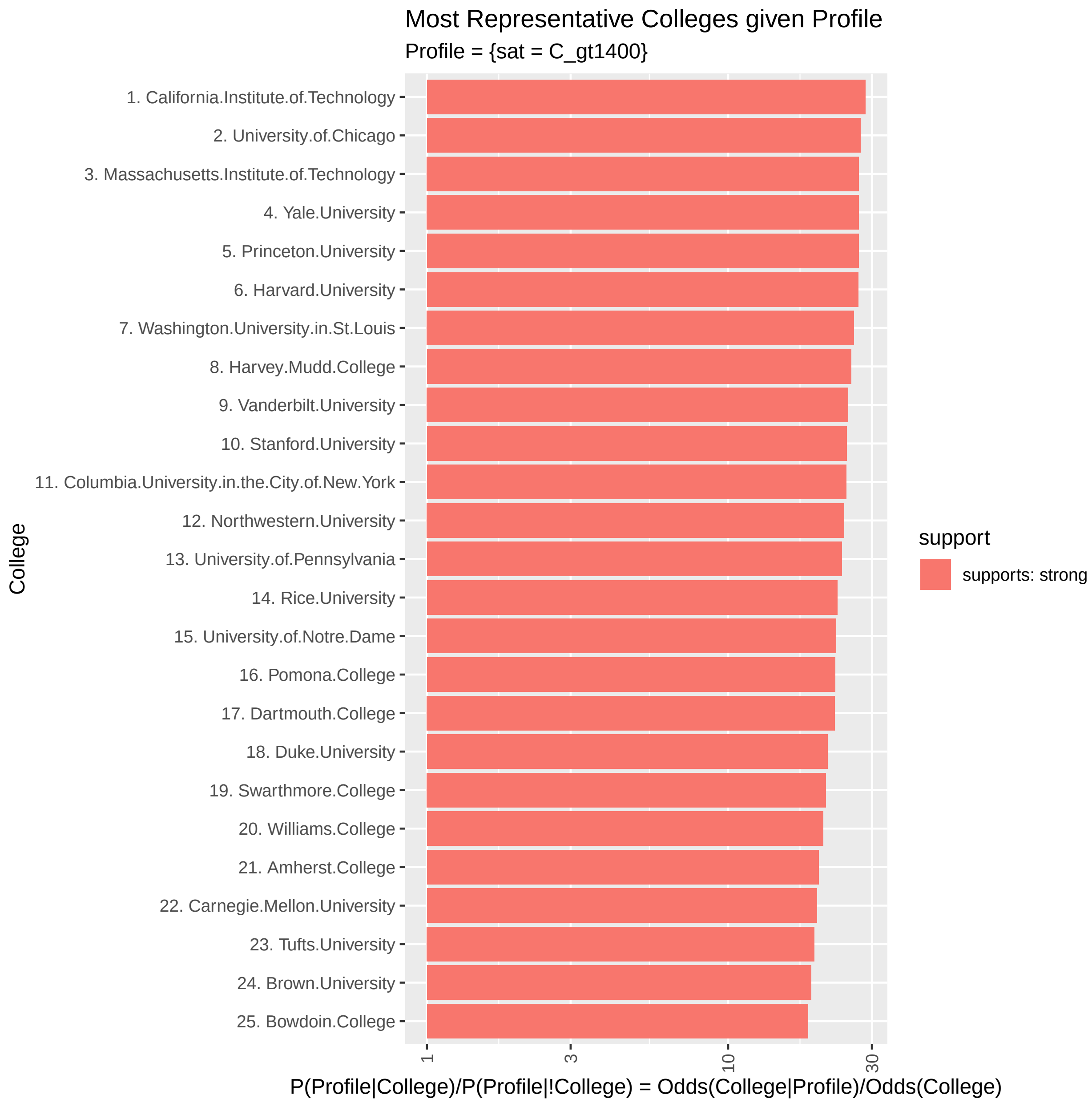
```
gbf_prof$gbf_min %>%
  dplyr::select(-matches("^(o|p)_|bur|College")) %>%
  {print(list("Minimal Explanation"= .))}
```

```
## $`Minimal Explanation`
## # A tibble: 1,089 x 3
##    hypothesis                                          gbf support
##    <fct>                                             <dbl> <fct>
##  1 College=X110404_California.Institute.of.Technolo~  28.5 supports: stro~
##  2 College=X144050_University.of.Chicago              27.5 supports: stro~
##  3 College=X166683_Massachusetts.Institute.of.Techn~  27.2 supports: stro~
##  4 College=X130794_Yale.University                    27.2 supports: stro~
##  5 College=X186131_Princeton.University               27.1 supports: stro~
##  6 College=X166027_Harvard.University                 27.0 supports: stro~
##  7 College=X179867_Washington.University.in.St.Louis  26.2 supports: stro~
##  8 College=X115409_Harvey.Mudd.College                25.6 supports: stro~
##  9 College=X221999_Vanderbilt.University              25.0 supports: stro~
## 10 College=X243744_Stanford.University                24.8 supports: stro~
## # ... with 1,079 more rows
```

```
college_data %>%
  arrange(desc(SAT_AVG)) %>%
  dplyr::select(1,starts_with("SAT")) %>%
  {print(list("By Highest Avg. SAT"=.))}
```

```
## $`By Highest Avg. SAT`
## # A tibble: 1,120 x 2
##    id                                                 SAT_AVG
##    <chr>                                                <dbl>
##  1 110404_California Institute of Technology             1534
##  2 144050_University of Chicago                          1504
##  3 166683_Massachusetts Institute of Technology          1503
##  4 166027_Harvard University                             1501
##  5 130794_Yale University                                1497
##  6 186131_Princeton University                           1495
##  7 115409_Harvey Mudd College                             1483
##  8 221999_Vanderbilt University                          1475
##  9 179867_Washington University in St Louis              1474
## 10 190150_Columbia University in the City of New York    1471
## # ... with 1,110 more rows
```

```
ev_str <- unlist(ev_list) %>% paste(names(.),.,sep=" = ",collapse="; ")
gbf_prof$gbf_min %>%
  filter(gbf>1) %>%
  top_n(25L,wt=gbf) %>%
  mutate(
    College=paste0(
      sprintf("%d. ",seq_along(College)),
      gsub("^.+_","",as.character(College))
    ) %>%
      factor(.,levels=rev(.))
  ) %>%
  {
    ggplot(.,aes(x=College,y=gbf,fill=support))+
      geom_bar(stat='identity')+
      scale_y_log10() +
      theme(axis.text.x = element_text(angle=90,hjust=1,vjust=0.5))+
      ylab("P(Profile|College)/P(Profile|!College) = Odds(College|Profile)/Odds(College)") +
      coord_flip() +
      ggtitle(
        "Most Representative Colleges given Profile",
        sprintf("Profile = {%s}",ev_str)
      )
  } %>%
  print()
```



Considering Only Ethnicity = Asian and Discipline = Social Sciences...

Now let's submit the student profile from Harvard as evidence and identify the most representative colleges:

$$\text{GBF}(h = \text{college } i; e = \{ethnicity = \text{Asian}, disc = \text{Social Science}\})$$

.

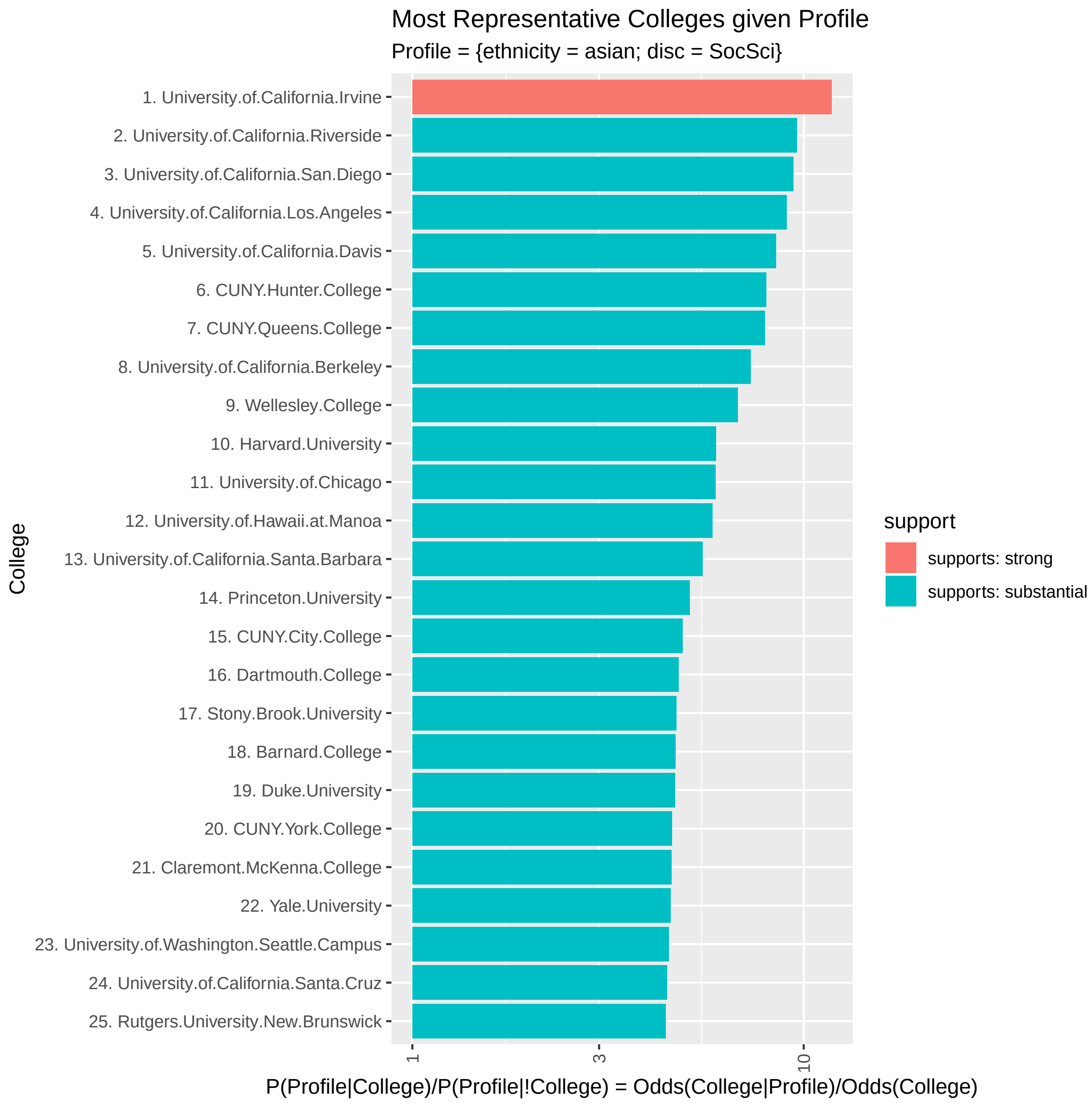
We see that Harvard comes in 10th, having only colleges in California and New York, with the exception of Wellesley College, with greater GBF.

```
ev_list  <- list(ethnicity="asian",disc="SocSci")
hyp_nms  <- "College"
gbf_prof1 <- bbn %>%
  gbf_all_hypcombos(
    ev_list  = ev_list,
    hyp_nms  = hyp_nms,
    phi_list = .$universe$levels[hyp_nms],
    verbose  = FALSE
  )
```

```
gbf_prof1$gbf_min %>%
  dplyr::select(-matches("^(o|p)_|bur|College")) %>%
  {print(list("Minimal Explanation"= .))}
```

```
## $`Minimal Explanation`
## # A tibble: 943 x 3
##    hypothesis                                       gbf support
##    <fct>                                          <dbl> <fct>
##  1 College=X110653_University.of.California.Irvi~ 11.8  supports: strong
##  2 College=X110671_University.of.California.Rive~  9.62 supports: substan~
##  3 College=X110680_University.of.California.San.~  9.41 supports: substan~
##  4 College=X110662_University.of.California.Los.~  9.04 supports: substan~
##  5 College=X110644_University.of.California.Davis  8.50 supports: substan~
##  6 College=X190594_CUNY.Hunter.College             8.02 supports: substan~
##  7 College=X190664_CUNY.Queens.College             7.95 supports: substan~
##  8 College=X110635_University.of.California.Berk~  7.33 supports: substan~
##  9 College=X168218_Wellesley.College               6.78 supports: substan~
## 10 College=X166027_Harvard.University              5.96 supports: substan~
## # ... with 933 more rows
```

```
ev_str <- unlist(ev_list) %>% paste(names(.),.,sep=" = ",collapse="; ")
gbf_prof1$gbf_min %>%
  filter(gbf>1) %>%
  top_n(25L,wt=gbf) %>%
  mutate(
    College=paste0(
      sprintf("%d. ",seq_along(College)),
      gsub("^.+_","",as.character(College))
    ) %>%
      factor(.,levels=rev(.))
  ) %>%
  {
    ggplot(.,aes(x=College,y=gbf,fill=support))+
      geom_bar(stat='identity')+
      scale_y_log10() +
      theme(axis.text.x = element_text(angle=90,hjust=1,vjust=0.5))+
      ylab("P(Profile|College)/P(Profile|!College) = Odds(College|Profile)/Odds(College)") +
      coord_flip() +
      ggtitle(
        "Most Representative Colleges given Profile",
        sprintf("Profile = {%s}",ev_str)
      )
  } %>%
  print()
```



With SAT > 1400...

Now, let's add in the criterion that SAT > 1400 ...

$$\text{GBF}(h = \text{college } i; e = \{SAT > 1400, ethnicity = \text{ Asian}, discipline = \text{Social Science}\})$$

.

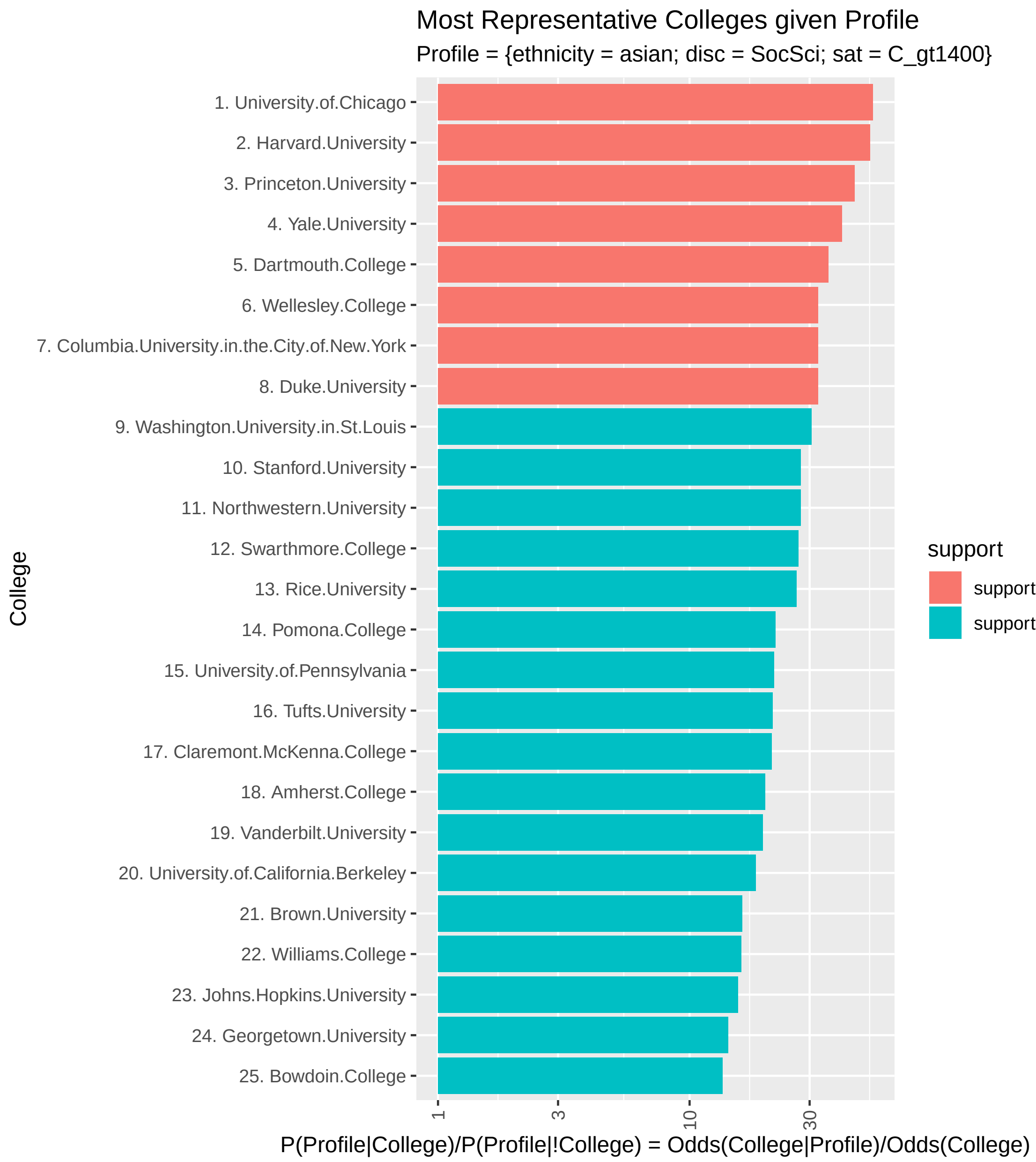
We see that Harvard jumps into a veritable tie for first with the University of Chicago.

```
ev_list  <- list(ethnicity="asian",disc="SocSci", sat = "C_gt1400")
hyp_nms  <- "College"
gbf_prof2 <- bbn %>%
  gbf_all_hypcombos(
    ev_list  = ev_list,
    hyp_nms  = hyp_nms,
    phi_list = .$universe$levels[hyp_nms],
    verbose  = FALSE
  )
```

```
gbf_prof2$gbf_min %>%
  dplyr::select(-matches("^(o|p)_|bur|College")) %>%
  {print(list("Minimal Explanation"= .))}
```

```
## $`Minimal Explanation`
## # A tibble: 1,096 x 3
##    hypothesis                                        gbf support
##    <fct>                                           <dbl> <fct>
##  1 College=X144050_University.of.Chicago            53.4 supports: very s~
##  2 College=X166027_Harvard.University               52.2 supports: very s~
##  3 College=X186131_Princeton.University             45.3 supports: very s~
##  4 College=X130794_Yale.University                  40.3 supports: very s~
##  5 College=X182670_Dartmouth.College                35.6 supports: very s~
##  6 College=X168218_Wellesley.College                32.5 supports: very s~
##  7 College=X190150_Columbia.University.in.the.Cit~  32.4 supports: very s~
##  8 College=X198419_Duke.University                  32.4 supports: very s~
##  9 College=X179867_Washington.University.in.St.Lo~  30.5 supports: strong
## 10 College=X243744_Stanford.University              27.6 supports: strong
## # ... with 1,086 more rows
```

```
ev_str <- unlist(ev_list) %>% paste(names(.),.,sep=" = ",collapse="; ")
gbf_prof2$gbf_min %>%
  filter(gbf>1) %>%
  top_n(25L,wt=gbf) %>%
  mutate(
    College=paste0(
      sprintf("%d. ",seq_along(College)),
      gsub("^.+_","",as.character(College))
    ) %>%
      factor(.,levels=rev(.))
  ) %>%
  {
    ggplot(.,aes(x=College,y=gbf,fill=support))+
      geom_bar(stat='identity')+
      scale_y_log10() +
      theme(axis.text.x = element_text(angle=90,hjust=1,vjust=0.5))+
      ylab("P(Profile|College)/P(Profile|!College) = Odds(College|Profile)/Odds(College)") +
      coord_flip() +
      ggtitle("Most Representative Colleges given Profile",sprintf("Profile = {%s}",ev_str))
  } %>%
  print()
```



References

Fitelson, Branden. 2007. "Likelihoodism, Bayesianism, and Relational Confirmation." *Synthese*. https://link.springer.com/content/pdf/10.1007/s11229-006-9134-9.pdf.

Good, I.J. 1985. "Weight of Evidence: A Brief Survey." *Bayesian Statistics 2; Bernardo, et Al. (Eds)*. https://www.cs.tufts.edu/~nr/cs257/archive/jack-good/weight-of-evidence.pdf.

Pacer, Michael, Joseph Williams, Xi Chen, Tania Lombrozo, and Thomas Griffiths. 2013. "Evaluating Computational Models of Explanation Using Human Judgment." *Proceedings of the 29th Conference on Uncertainty in Artificial Intelligence (Uai2013)*. https://arxiv.org/ftp/arxiv/papers/1401/1401.3893.pdf.

Tenenbaum, Joshua B., and Thomas L. Griffiths. 2001. "The Rational Basis of Representativeness." *Proceedings of the 23rd Annual Conference of the Cognitive Science Society*. http://web.mit.edu/cocosci/

Papers/cogsci01_final.pdf.

Yuan, Changhe, Heejin Lim, and Tsai-Ching Lu. 2011. "Most Relevant Explanation in Bayesian Networks." *Journal of Artificial Intelligence Research*. https://arxiv.org/ftp/arxiv/papers/1401/1401.3893.pdf.