The person-environment fit of immigrants to the United States: A registered report

Supplemental materials

2024-01-01

Contents

List of Tables

List of Figures

1 Workspace

1.1 Packages and functions

```
library(here) # for file management
library(knitr)
library(xfun)
library(tidyverse) # for cleaning
library(mirt) # for scoring narrow traits
library(sjPlot) # figures
library(modelsummary) # pretty tables
library(papaja) # for number formatting
library(kableExtra) #also pretty tables
library(psych) # for internal consistency
library(ggridges) # for ridgeline figures
library(ggpubr) # for nice looking figures
library(ggdist) # make it rain
library(viridis) # for color-blind friendly palette
library(lme4) # for multilevel modeling
library(lmerTest)
library(car) # for ANOVA test
library(emmeans) # for estimating slopes
library(marginaleffects) # for estimating slopes
library(maps) # for US map
library(censusapi) # for US census API
library(broom)
library(broom.mixed)
library(MetBrewer) # colors!
library(whitestrap) # Heteroskedasticity test
library(showtext)
library(zipcodeR)
# set options for model summary package
options(modelsummary_get = "easystats")

# get pretty theme for figures
source(here("supplemental/zz_clean-theme.R"))
```

1.2 API

Data preparation includes downloading datasets prepared by the US Census. For this, we rely on the US Census API. If you wish to reproduce this part of the script, you will need an API key. You can request a key at https://api.census.gov/data/key_signup.html. Once received, create a .R script with the following code:

```
census_key = "Your Key Here"
```

Save this file as `census-api-key.R` in your working directory.

1.3 Data

Here we load in the raw data. At the time of writing this script, data were not publicly available, although we anticipate their release on the Harvard Datavere in the future. The Rdata file contains several important objects. Relevant to these analyses are:

- `SAPAdata07feb2017thru18aug2023` – a data frame, the data collected via SAPA between February 17, 2017 and August 18, 2023 (the date of Stage 1 acceptance).
- `itemsSPI` – a character vector, the names of all items in the SPI-135
- `keys.list` – a list containing the names of scales included in the SAPA item bank and the relative scoring (i.e., 1, -1, or 0) of each item onto each scale.

```r
# these data were previously published on Dataverse, minus the zipcodes, which were
# provided by the SAPA data owner
load(here("../../../../Dropbox (University of Oregon)/work/research/research PIElab/SAPA data/original 
)
sapa = SAPAdata07feb2017thru18aug2023
rm(SAPAdata07feb2017thru18aug2023,
   #extra objects loaded with data -- unnecessary for these analsyes
   cuts020717, cuts022521, cuts050118, cuts071020,
   cuts071517, cuts091818, cuts111119)
```

```r
# here we specify the items that belong in the Ryff well-being scales
ryff = str_c("q_", c(4983:5024))
```

Notably, the original data file is extremely large (N > 2 million) and nearly impossible to work with. However, after removing participants who do not report a zip code, the dataset is more manageable (N ~ 425,000).

```r
sapa = sapa %>%
  filter(!is.na(ZCTA)) # must have non-missing zip code

sapa = sapa %>%
  select(RID, english, sex, ZCTA, country,
         age, marstatus, ethnic,
         education, p1edu, p2edu, ZCTAstate,
         health, stress,
         all_of(itemsSPI), #this vector comes with the original data
         all_of(ryff)) %>%
  filter(age >= 13)
```

2 Cleaning

We first calculate for each participant the number of SPI items they completed.

```r
sapa = sapa %>%
  mutate(num_SPI = rowSums(!is.na(select(., all_of(itemsSPI)))))
```

We also create the `age_group` variable for future analysis.

```
sapa = sapa %>%
  mutate(age_group = case_when(
    age < 18 ~ "13-17",
    age < 26 ~ "18-25",
    age < 36 ~ "26-35",
    age < 46 ~ "36-45",
    age < 56 ~ "46-55",
    age < 66 ~ "56-65",
    TRUE ~ "66+"
  ))
```

Here we identify the geographic region of the world each participant was born in.

```
sapa = sapa %>%
  mutate(region = case_when(
    country %in% c("ARM","AZE","BHR","CYP","GEO","IRN","IRQ","ISR",
                   "JOR","KWT","LBN","OMN","PSE","QAT","SAU","SYR",
                   "TUR","ARE","YEM","AFG","KAZ","KGZ","TJK","TKM",
                   "UZB")                                           ~ "Middle East",
    country %in% c("CHN","HKG","MAC","JPN","PRK","KOR","MNG","TWN") ~ "East Asia",
    country %in% c("BGD","BTN","XXX","IND","MDV","NPL","PAK","LKA",
                   "BRN","KHM","IDN","LAO","MYS","MMR","PHL","SGP",
                   "THA","TLS","VNM","IOT")                         ~ "South/Southeast Asia",
    country %in% c("DZA","AGO","BEN","BWA","BFA","BDI","CMR","CPV",
                   "CAF","TCD","COM","COD","DJI","EGY","GNQ","ERI",
                   "ETH","GAB","GMB","GHA","GNB","GIN","CIV","KEN",
                   "LSO","LBR","LBY","MDG","MWI","MLI","MRT","MUS",
                   "MAR","MOZ","NAM","NER","NGA","REU","RWA","SHN",
                   "STP","SEN","SYC","SLE","SOM","ZAF","SDN","SWZ",
                   "TZA","TGO","TUN","UGA","ESH","ZMB","ZWE","COG") ~ "Africa",
    country %in% c("ALB","AND","AUT","BLR","BEL","BIH","BGR","HRV",
                   "CZE","DNK","EST","FRO","FIN","FRA","DEU","GIB",
                   "GRC","GRL","HUN","ISL","IRL","ITA","UNK","LVA",
                   "LIE","LTU","LUX","MKD","MLT","MDA","MCO","MNE",
                   "NLD","NOR","POL","PRT","ROU","RUS","RSM","SRB",
                   "SVK","SVN","ESP","SWE","CHE","UKR","GBR","VAT") ~ "Europe",
    country %in% c("MEX","AIA","ATG","ARG","ABW","BHS","BRB","BLZ",
                   "BMU","BOL","BRA","VGB","CYM","CHL","COL","COK",
                   "CRI","CUB","DMA","DOM","ECU","SLV","FLK","GUF",
                   "GRD","GLP","GTM","GUY","HTI","HND","JAM","MTQ",
                   "MSR","ANT","NIC","PAN","PRY","PER","PRI","KNA",
                   "LCA","VCT","SUR","TTO","VIR","URY","VEN")       ~ "South and Central America",
    country %in% c("AUS","FSM","FJI","PYF","GUM","KIR","MHL","NRU",
                   "NCL","NZL","NIU","NFK","MNP","PLW","PNG","PCN",
                   "WSM","SLB","TKL","TON","TUV","VUT","WLF")       ~ "Oceania",
    country %in% c("CAN","JTN","MID","SPM","TCA","WAK")             ~ "North America",
    TRUE                                                            ~ NA_character_
    )
  )
```

Here we identify the official language of the country of origin. We restrict categorization to languages that are the official language for five or more countries; all others are listed as "other". In cases when the country has more than one official language, the language we use will be the most common official language. For example, both English and French are official languages of Canada; English was chosen as the designation because English is an official language in 59 countries while French is the official language in only 29.

```
sapa = sapa %>%
  mutate(language = case_when(
    country %in% c("BHS", "BRB", "BLZ", "BWA", "BRN", "BDI",
                   "CMR", "CAN", "DMA", "SWZ", "FJI", "GMB",
                   "GHA", "GRD", "GUY", "IND", "IRL", "JAM",
                   "KEN", "KIR", "LSO", "LBR", "MWI", "MLT",
                   "MHL", "MUS", "FSM", "NAM", "NRU", "NGA",
                   "PAK", "PLW", "PNG", "PHL", "RWA", "KNA",
                   "LCA", "VCT", "WSM", "SYC", "SLE", "SGP",
                   "SLB", "ZAF", "SSD", "SDN", "TZA", "TON",
                   "TTO", "TUV", "UGA", "VUT", "ZMB", "ZWE",
                   "ATG", "AUS", "NZL", "GBR", "USA") ~ "English",
    country %in% c("BEN", "BFA", "COG", "COD", "CIV", "FRA",
                   "GAB", "GIN", "MLI", "MCO", "NGA", "SEN",
                   "TGO", "BEL", "BDI", "CMR", "CAN", "TCD",
                   "CAF", "COM", "DJI", "GNQ", "HTI", "LUX",
                   "MDG", "RWA", "SYC", "CHI", "VUT") ~ "French",
    country %in% c("DZA", "BHR", "TCD", "COM", "DJI",
                   "EGY", "IRQ", "JOR", "KWT", "LBN",
                   "LBY", "MRT", "MAR", "OMN", "PSE",
                   "QAT", "SAU", "SOM", "SDN", "SYR",
                   "TUN", "ARE", "YEM") ~ "Arabic",
    country %in% c("MEX", "COL", "ESP", "ARG", "PER",
                   "VEN", "CHL", "GTM", "ECU", "BOL",
                   "CUB", "DOM", "HND", "PRY", "SLV",
                   "NIC", "CRI", "PAN", "URY", "GNQ") ~ "Spanish",
    country %in% c("AGO", "BRA", "CPV", "TLS", "GNQ",
                   "GNB", "MOZ", "PRT", "STP") ~ "Portuguese",
    country %in% c("DEU", "AUT", "CHE", "BEL", "LUX", "LIE") ~ "German",
    country %in% c("RUS", "BLR", "KAZ", "KGZ", "TJK") ~ "Russian",
    country %in% c("COD", "KEN", "RWA", "TZA", "UGA") ~ "Swahili",
    TRUE                                                        ~ "OTHER"
    )
  )
```

We identify the highest degree between the two parents reported and save this to a new variable.

Change from preregistration: Pilot data recorded education as a number and we converted this to a factor. In the present data, education is already an ordered factor.

```
# find max parent edu
sapa$parent_edu = apply(sapa[,c("p1edu", "p2edu")], 1, max, na.rm=T)
```

2.1 ZCTA information

Using data taken from the US Census (ACS 5-Year Estimates Data Profiles from 2019), we calculate the proportion of residents in each zipcode who are immigrants (labeled “foreign-born”). Data can be obtained through the US Census API. See (#api) for information about obtaining and using the API key in this file.

```r
source(here("scripts/census-api-key.R"))
Sys.setenv(CENSUS_KEY=census_key)

# list APIs available
# apis <- listCensusApis()

#from relevant API, list available variables
# vars = listCensusMetadata(
#   name = "acs/acs5/profile",
#   vintage = 2019,
#   type = "variables")

uscensus <- getCensus(
    name = "acs/acs5/profile",
    vars = c("NAME",
             "DP05_0001E",  # number of persons
             "DP02_0104E",  # number of foreign-born
             "DP02_0068PE", # proportion of residents with college degree
             "DP02_0093PE", # proportion of foreign-born residents,
             "DP03_0128PE"  # proportion of people living below poverty level
             ),
    region = "zip code tabulation area", regionin = "state:*",
    vintage = 2019)

uscensus = uscensus %>%
  # rename the variables we need
  rename(ZCTA = zip_code_tabulation_area,
         population    = DP05_0001E,
         fb_pop        = DP02_0104E,
         prop_college = DP02_0068PE,
         prop_foreign = DP02_0093PE,
         prop_poverty = DP03_0128PE)  %>%
  select(ZCTA, population, fb_pop, prop_college, prop_foreign, prop_poverty) %>%
  filter(ZCTA %in% sapa$ZCTA) %>%
  # we remove the value "null" from all variables
  mutate(across(everything(), ~ifelse(. == "null", NA, .))) %>%
  # make both variables number
  mutate(across(everything(), as.numeric)) %>%
  # remove values indicating missingness
  mutate(across(everything(), ~ifelse(. < 0, NA, .))) %>%
  # we only use ZCTAs with non missing data
  filter(!is.na(population) &
         !is.na(fb_pop) &
         !is.na(prop_college) &
         !is.na(prop_foreign))


# merge with sapa data frame
sapa = left_join(sapa, uscensus)
```

In the following section, we extract from the same census files several variables related to the ancestry and place-of-origin of residents. Unfortunately, the only variable in the US Census data that captures information relevant to all US immigrants assesses continent of origin, which has little specificity. Two additional variables

– ancestry and country-of-origin – do not assess all possible locales, and so only apply to some participants. Importantly, some variables describe immigrant populations while others describe all ZCTA residents. In all cases, both US Census and SAPA categories required some collapsing to match across datasets.

Table S1: ZCTA match variables

Variable	Label	Geographies	Population
Continent of origin	`origin`	Europe, Asia, Africa, Oceania, Latin America, North America	Foreign-born residents
Ancestry	`ancestry`	Scotland, France, Czech Republic, Poland, Slovakia, (French) Canadian, Denmark, sub-Saharan Africa, Germany, West Indies (excluding Hispanic/Latino), Netherlands, Greece, Sweden, England, Switzerland, Hungary, Ukraine, Ireland, Wales, Italy, Russia, Lithuania, Scotch-Irish, Arab, Norway	All residents
Country of origin	`originCountry`	China, Cuba, Dominican Republic, El Salvador, Guatemala, India, Mexico, Philippines, Vietnam, All other countries	Foreign-born residents

```r
uscensus_origin <- getCensus(
    name = "acs/acs5/profile",
    vars = c("NAME",
             "DP02_0105PE", "DP02_0106PE", "DP02_0107PE", "DP02_0108PE",
             "DP02_0109PE", "DP02_0110PE"
             ),
    region = "zip code tabulation area", regionin = "state:*",
    vintage = 2019)

uscensus_origin = uscensus_origin %>%
  select(-state, -NAME) %>%
  # rename the variables we need
  rename(ZCTA = zip_code_tabulation_area,
         origin_Europe = DP02_0105PE,
         origin_Asia = DP02_0106PE,
         origin_Africa = DP02_0107PE,
         origin_Oceania = DP02_0108PE,
         origin_LatinAm = DP02_0109PE,
         origin_NorthAm = DP02_0110PE
         )  %>%
  # we remove the value "null" from all variables
  mutate(across(everything(), ~ifelse(. == "null", NA, .))) %>%
  # make all variables number
  mutate(across(everything(), as.numeric)) %>%
  # remove values indicating missingness
  mutate(across(everything(), ~ifelse(. < 0, NA, .)))

uscensus_origin = uscensus_origin %>%
  pivot_longer(starts_with("origin_"), names_to = "origin_continent", values_to = "prop_originContinent
  mutate(origin_continent = str_remove(origin_continent, "^origin_"))
```

```r
# in the SAPA data, we merge some categories through renaming (to match the census categories)

sapa = sapa %>%
  mutate(
    origin_continent = case_when(
    region %in% c("Middle East", "Africa") ~ "Africa",
    str_detect(region, "Asia") ~ "Asia",
    region == "South and Central America" ~ "LatinAm",
    region == "North America" ~ "NorthAm",
    TRUE ~ region
  ))

# merge with sapa data frame
sapa = left_join(sapa, uscensus_origin)
```

```r
uscensus_ancestry <- getCensus(
    name = "acs/acs5/profile",
    vars = c("NAME", "DP02_0143PE", "DP02_0130PE", "DP02_0126PE", "DP02_0139PE",
             "DP02_0144PE", "DP02_0131PE", "DP02_0127PE", "DP02_0145PE",
             "DP02_0132PE","DP02_0150PE", "DP02_0128PE", "DP02_0133PE",
             "DP02_0146PE", "DP02_0129PE", "DP02_0147PE", "DP02_0134PE",
             "DP02_0148PE", "DP02_0135PE","DP02_0140PE", "DP02_0149PE",
             "DP02_0136PE", "DP02_0141PE", "DP02_0137PE", "DP02_0142PE",
             "DP02_0125PE","DP02_0138PE"
             ),
    region = "zip code tabulation area", regionin = "state:*",
    vintage = 2019)

uscensus_ancestry = uscensus_ancestry %>%
  select(-state, -NAME) %>%
  # rename the variables we need
  rename(ZCTA = zip_code_tabulation_area,
         ancestry_SCT = DP02_0143PE, # scotland -- change to GBR
         ancestry_FRA = DP02_0130PE, # france
         ancestry_CZE = DP02_0126PE, # czech republic
         ancestry_POL = DP02_0139PE, # poland
         ancestry_SVK = DP02_0144PE, # slovakia
         ancestry_CAN = DP02_0131PE, # (french) canadian
         ancestry_DNK = DP02_0127PE, # denmark
         ancestry_ssa = DP02_0145PE, # sub-saharan africa -- change in sapa
         ancestry_DEU = DP02_0132PE, # germany
         ancestry_wei = DP02_0150PE, # west indies excl hispanic origin -- change in sapa
         ancestry_NLD = DP02_0128PE, # netherlands
         ancestry_GRC = DP02_0133PE, # greece
         ancestry_SWE = DP02_0146PE, # sweden
         ancestry_ENG = DP02_0129PE, # england -- change to GBR
         ancestry_CHE = DP02_0147PE, # switzerland
         ancestry_HUN = DP02_0134PE, # hungary
         ancestry_UKR = DP02_0148PE, # ukraine
         ancestry_IRL = DP02_0135PE, # ireland
         ancestry_PRT = DP02_0140PE, # portugal
         ancestry_WEL = DP02_0149PE, # welsh -- change to GBR
         ancestry_ITA = DP02_0136PE, # italy
```

```
         ancestry_RUS = DP02_0141PE, # russia
         ancestry_LTU = DP02_0137PE, # lithuania
         ancestry_SCI = DP02_0142PE, # scotch-irish -- change to GBR
         ancestry_arb = DP02_0125PE, # arab, change in SAPA
         ancestry_NOR = DP02_0138PE  # norway
         )  %>%
  # we remove the value "null" from all variables
  mutate(across(everything(), ~ifelse(. == "null", NA, .))) %>%
  # make all variables number
  mutate(across(everything(), as.numeric)) %>%
  # remove values indicating missingness
  mutate(across(everything(), ~ifelse(. < 0, NA, .)))

uscensus_ancestry = uscensus_ancestry %>%
  pivot_longer(starts_with("ancestry_"),
               names_to = "ancestry_country",
               values_to = "prop_ancestry") %>%
  mutate(ancestry_country = str_remove(ancestry_country, "^ancestry_"))%>%
  mutate(ancestry_country = case_when(
    ancestry_country %in% c("SCT", "ENG", "WEL", "SCI") ~ "GBR",
    TRUE ~ ancestry_country
  ))

# in the SAPA data, we merge some categories
# through renaming (to match the census categories)
# note that some countries could be included in multiple
# ancestry categories: Comoros, Sudan, and Somalia are both # sub-saharan arab countries. We chose to i
# the sub-saharan African category

sapa = sapa %>%
  mutate(country = as.character(country),
    ancestry_country = case_when(
    country %in% c(
      # angola, benin, botswana, burkina faso, burundi,
      # cabo verde, cameroon, central african republic,
      # chad, comoros,
      "AGO", "BEN", "BWA", "BFA", "BDI", "CPV", "CMR",
      "CAF", "TCD", "COM",
      # congo, dem. repub, congo, rep, cote d'ivoire,
      # equatorial guinea, eritrea, ethiopia, gabon,
      # gambia, tha, ghana, guinea,
      "COG", "COD", "CIV", "GNQ", "ERI", "ETH", "GAB",
      "GMB", "GHA", "GIN",
      # guinea-bissau, kenya, lesotho, liberia,
      # madagascar, malawi, mali, mauritania, mauritius,
      # mozambique,
      "GNB", "KEN", "LSO", "LBR", "MDG", "MWI", "MLI",
      "MRT", "MUS", "MOZ",
      # namibia, niger, nigeria, rwanda, sao tome
      # and principe, senegal, seychelles, sierra leone,
      # somalia, south africa, south sudan,
      "NAM", "NER", "NGA", "RWA", "STP", "SEN", "SYC",
      "SLW", "SOM", "ZAF", "SSD",
```

```
      # sudan, tanzania, togo, uganada, zambia, zimbabwe
      "SDN", "TZA", "TGO", "UGA", "ZMB", "ZWE"  )           ~ "ssa",
    country %in% c(
      # cuba, jamaica, puerto rico, antigua, st kitts
      # and nevis, st martin, anguilla, martinique,
      # dominica, st lucia, bermuda, trinidad and tobago,
      # suriname, guyana, virgin islands
      "CUB", "JAM", "PRI", "ATG", "KNA", "MAF", "AIA",
      "MTQ", "DMA", "LCA", "BMU", "TTO", "SUR",  "GUY",
      "VGB") ~                                               "wei",
    country %in% c(
      # algeria, bahrain, djibouti, egypt, iraq, jordan,
      # kuwait, lebanon, libya, mauritania, morocco,
      # oman, palestine, qatar, saudi Arabia, syria, tunisia,
      # united Arab Emirates, yemen
      "DZA", "BHR", "DJI", "EGY", "IRQ", "JOR", "KWT",
      "LBN", "LBY", "MRT", "MAR", "OMN", "PSE", "QAT",
      "SAU", "SYR", "TUN", "ARE", "YEM" ) ~                  "arb",
    TRUE ~ country # keep all others
  ))


# merge with sapa data frame
sapa = left_join(sapa, uscensus_ancestry)
```

```
uscensus_originCountry <- getCensus(
    name = "acs/acs5",
    vars = c("NAME",
             "B05015_002E", "B05015_005E", "B05015_008E",
             "B05015_011E", "B05015_014E", "B05015_017E",
             "B05015_020E", "B05015_023E", "B05015_026E",
             "B05015_029E"
             ),
    region = "zip code tabulation area", regionin = "state:*",
    vintage = 2019)

uscensus_originCountry = uscensus_originCountry %>%
  select(-state, -NAME) %>%
  # rename the variables we need
  rename(ZCTA = zip_code_tabulation_area,
         originCountry_China = B05015_002E,
         originCountry_Cuba = B05015_005E,
         originCountry_DominicanR = B05015_008E,
         originCountry_ESalvador = B05015_011E,
         originCountry_Guatemala = B05015_014E,
         originCountry_India = B05015_017E,
         originCountry_Mexico = B05015_020E,
         originCountry_Philippines = B05015_023E,
         originCountry_Vietnam = B05015_026E,
         originCountry_Other = B05015_029E
         )  %>%
  # we remove the value "null" from all variables
  mutate(across(everything(), ~ifelse(. == "null", NA, .))) %>%
```

```
  # make all variables number
  mutate(across(everything(), as.numeric)) %>%
  # remove values indicating missingness
  mutate(across(everything(), ~ifelse(. < 0, NA, .)))

uscensus_originCountry = uscensus_originCountry %>%
  pivot_longer(starts_with("originCountry_"),
               names_to = "originCountry_country",
               values_to = "match_originCountry") %>%
  mutate(originCountry_country = str_remove(originCountry_country, "^originCountry_"))
# in the SAPA data, we merge some categories through renaming (to match the census categories)

sapa = sapa %>%
  mutate(country = as.character(country),
    originCountry_country = case_when(
    country %in% c("CHN", "TWN") ~ "China",
         country == "CUB" ~ "Cuba",
         country == "DOM" ~ "DominicanR",
         country == "SLV" ~ "ESalvador",
         country == "GTM" ~ "Guatemala",
         country == "IND" ~ "India",
         country == "MEX" ~ "Mexico",
         country == "PHL" ~ "Philippines",
         country == "VNM" ~ "Vietnam",
         TRUE ~ "Other"
  ))


# merge with sapa data frame
sapa = left_join(sapa, uscensus_originCountry) %>%
  # change from estimate to percentage
  mutate(match_originCountry = map2_dbl(match_originCountry,
                                        population, ~100*.x/.y))

# Finally, we have to adjust the continent-of-origin and country-of-origin variable to
# reflect the proportion of all ZCTA residents, not only the foreign-born residents

sapa = sapa %>%
  mutate(prop_originContinent = (prop_originContinent*(fb_pop/population)),
         match_originCountry = (match_originCountry*(fb_pop/population)))
```

We also categorize the region of the United States associated with each ZCTA.

```
sapa = sapa %>%
  mutate(zcta_region = case_when(
    ZCTAstate %in% c("MA", "ME", "NH", "VT", "RI", "CT") ~ "New England",
    ZCTAstate %in% c("NY", "PA", "NJ", "DE", "MD", "DC") ~ "Mid-Atlantic",
    ZCTAstate %in% c("AR", "LA", "KY", "TN", "MS", "AL",
                     "GA", "FL", "SC", "NC", "WV", "VA") ~ "Southeast",
    ZCTAstate %in% c("MT", "ID", "WY", "NV", "UT", "CO") ~ "Rocky Mountains",
    ZCTAstate %in% c("ND", "SD", "NE", "KS", "MN", "IA",
                     "MO", "WI", "IL", "MI", "OH", "IN") ~ "Midwest",
    ZCTAstate %in% c("TX", "AZ", "NM", "OK") ~ "Southwest",
```

```
    ZCTAstate %in% c("WA", "OR", "CA", "AK", "HI") ~ "Pacific Coastal"
  ))
```

2.2 Filter participants

We create three data frames. The first contains immigrants to the United States. To be included, a participant must meet the following criteria:

- Have a non-missing country of origin that is not the United States
- Have a non-missing zipcode that is in the United States
- Not claim their zipcode is 90210
- Complete at least 50 personality items.

```
immigrants = sapa %>%
  filter(country != "USA") %>%
  filter(!is.na(ZCTA)) %>%
  filter(ZCTA != 90210) %>%
  filter(num_SPI >= 50)
```

For the main analyses, we also select on the basis that participants speak English “Well” or “Very Well” This is a quality control issues – we recognize that it limits the generalizability of findings to all immigrants, but we worry that data from those who speak English poorly will have greater measurement error than signal. As exploratory analysis, we will test whether English fluency moderates fit. More specifically, we expect low fluency to be associated with worse fit, although we will not be able to distinguish the effects of measurement error from poor fit due to language barriers.

```
# df that includes all immigrants
immigrants_all = immigrants
# df that only includes immigrants with fluency
immigrants = immigrants %>%
  filter(english %in% c("Well", "VeryWellFluentNative"))
```

Next we create a data frame of native US residents. To be included, a participant must meet the following criteria:

- Have a non-missing country of origin that is the United States
- Have a non-missing zipcode that is in the United States
- Not claim their zipcode is 90210
- Speak English “Well” or “Very Well”.

```
us_data = sapa %>%
  filter(country == "USA") %>%
  filter(!is.na(ZCTA)) %>%
  filter(ZCTA != 90210) %>%
  filter(english %in% c("Well", "VeryWellFluentNative")) %>% # not in Dataverse files
  filter(num_SPI >= 50)
```

```
save(immigrants, immigrants_all, us_data, # data
     keys.list, # information about SPI items
     file = here("data/precleaned.Rdata"))

save(immigrants_all, file = here("data/immigrants-all.Rdata"))
```

2.3 Personality traits

In the immigrant data set, we score both the broad (5) and narrow (27) SPI traits. Broad traits are scored using a sum-score (average response) approach, as befits multidimensional scales. Narrow traits, on the other hand, are unidimensional and may be scored using item response theory. Here, we score responses using IRT parameters derived from a separate sample of more than 500,000 participants (see Condon -Condon (2017) for more information; IRT parameters are derived from the combined Replication and Confirmatory samples).

```
# spi scales (32 in total)
spi.keys = keys.list[grepl("SPI_135_27_5", names(keys.list))]
# broad (big five) scales
broad_scored = scoreItems(spi.keys[1:5],
                          immigrants,
                          impute = "none")
# reliability
(b5_alpha = round(broad_scored$alpha,3))
# add scores to data
scores_5 = broad_scored$scores
immigrants = cbind(immigrants, scores_5)


# broad (big five) scales
broad_scored = scoreItems(spi.keys[1:5],
                          us_data,
                          impute = "none")
scores_5 = broad_scored$scores
us_data = cbind(us_data, scores_5)
```

```
IRT_model_path = here("data/IRT_model_list_SPI_135.RData")

# load the IRT models
load(IRT_model_path)

# custom scoring function
source(here("scripts/fun_score-spi.r"))

# Now, use the custom function to score the data.
immigrants = score_SPI_IRT_all(immigrants, t_score = FALSE)

immigrants = immigrants %>%
  rename_all(~str_remove(., "135_27_5_")) %>% # shorten names of traits
  rename(wellbeing = WellBeing)

us_data = score_SPI_IRT_all(us_data, t_score = FALSE)
```

We also score traits in the US data for later sensitivity analyses.

```
us_data = score_SPI_IRT_all(us_data, t_score = FALSE)
```

Well-being items are removed from the profile similarity assessment because the well-being scale will be used as an outcome.

```
wb_items = c("q_578", "q_811", "q_2765", "q_820", "q_1371")

immigrants = immigrants %>%
  select(-all_of(wb_items))

us_data = us_data %>%
  select(-all_of(wb_items))
```

2.4 Psychological Well-being

Here we calculate scores for the Psychological Well-Being Scale (Ryff and Keyes (1995)) and count the number of participants with valid scores. If the minimum number of pairwise administrations of these items is at least 300, we will report analyses using these scales in the manuscript; if the pairwise administrations are under 300, we will still conduct the analysis, but not report these in text.

```
n_pair_ryff = immigrants %>%
  select(any_of(ryff)) %>%
  psych::pairwiseCount()

min(n_pair_ryff)
```

```
## [1] 4
```

```
max(n_pair_ryff)
```

```
## [1] 302
```

```
mean(n_pair_ryff)
```

```
## [1] 22.32029
```

Next, we calculate scores on the well-being measure.

```
ryff.keys = list(
  ryff_autonomy        = c("q_4988", "-q_4997",  "q_4993",  "q_5023", "-q_5004",  "q_5010", "-q_5013"),
  ryff_environ_mastery = c("q_4984", "-q_5020", "-q_4989",  "q_5024", "-q_4994",  "q_4998", "-q_5000"),
  ryff_personal_growth = c("q_5017", "-q_4985", "-q_5021",  "q_5001", "-q_5014", "-q_5005",  "q_5011"),
  ryff_pos_relations   = c("q_5009",  "q_4983", "-q_5012", "-q_4987",  "q_5019", "-q_4992",  "q_4996"),
  ryff_purpose         = c("q_5008", "-q_4990",  "q_4999",  "q_5015", "-q_5006", "-q_5018", "-q_5002"),
  ryff_self_accept     = c("-q_5007", "q_4986", "-q_5016",  "q_4991",  "q_5022", "-q_4995",  "q_5003")
)

ryff_scored = scoreItems(ryff.keys,
                         immigrants,
                         impute = "none")
# reliability
(ryff_alpha = round(ryff_scored$alpha,3))
```

```
##       ryff_autonomy ryff_environ_mastery ryff_personal_growth
## alpha         0.799                 0.63                0.895
##       ryff_pos_relations ryff_purpose ryff_self_accept
## alpha              0.676        0.744            0.856
```

Table S2: Correlations between Ryff Psychological Well-being scales

	autonomy	environ_mastery	personal_growth	pos_relations	purpose	self_accept
autonomy	1.00	0.10	0.16	0.03	0.13	0.07
environ_mastery	0.10	1.00	0.29	0.26	0.22	0.32
personal_growth	0.16	0.29	1.00	0.23	0.17	0.31
pos_relations	0.03	0.26	0.23	1.00	0.13	0.36
purpose	0.13	0.22	0.17	0.13	1.00	0.28
self_accept	0.07	0.32	0.31	0.36	0.28	1.00

```r
# add scores to data
scores_ryff = ryff_scored$scores
immigrants = cbind(immigrants, scores_ryff)

immigrants = immigrants %>%
  select(-all_of(ryff))

us_data = us_data %>%
  select(-all_of(ryff))
```

At this point, we will examine the correlations between the well-being scales to determine whether they are distinct or all fall from a general factor. If the latter, we will aggregate into a single well-being score; otherwise, we will examine each separately in the well-being analysis section (RQ4) of this file.

```r
scores_ryff %>%
  as.data.frame() %>%
  rename_all(str_remove,"ryff_") %>%
  cor(use = "pairwise") %>%
  round(2) %>%
  kable(
    booktabs = T,
    caption = "Correlations between Ryff Psychological Well-being scales"
  ) %>%
  kable_styling()
```

2.5 Regional profiles

2.5.1 US ZCTAs

Here we calculate the profile of each ZCTA, by calculating the average response to each item. We also ipsatize, or calculate the standardized score within each ZCTA.

```r
#z-score
z_fun = function(x){
  y = (x-mean(x, na.rm=T))/(sd(x, na.rm=T))
  return(y)
}
```

```
zip_profiles = us_data %>%
  group_by(ZCTA) %>%
  summarise(
    across( starts_with("q_"), \(x) mean(x, na.rm=T)), # mean of each personality item
    n_zip = n() # number of participants
  ) %>%
  ungroup() %>%
  filter(n_zip >= 30)


zip_profile_long = zip_profiles %>%
  #transform to long form
  pivot_longer(
    # name of column storing variable names
    names_to = "item",
    # name of column storing values
    values_to = "zip_response",
    # which variables to stack
    cols = starts_with("q_"),
    # remove rows with missing values
    values_drop_na = T)

# %>%
  # select(ZCTA, item, zip_response)
```

2.6 Long-form dataset

First we identify and remove duplicate rows.

```
dupe_rows = janitor::get_dupes(immigrants)
immigrants = immigrants %>%
  with_groups(RID, filter, row_number() == 1)
```

To estimate person-environment fit, we require data to be long form, specifically one row per item per participant. For each immigrant and row, we include the participant's response, the average response of their zipcode, and the average response of their country of origin.

```
immigrants_profile = immigrants %>%
  pivot_longer(
    cols = starts_with("q_"),
    names_to = "item",
    values_to = "response",
    values_drop_na = T
  )

immigrants_profile = immigrants_profile %>%
  #merge with location profiles
  left_join(zip_profile_long) %>%
  # standardize responses within profile
  with_groups(RID, mutate, across(
    # mutate these columns
    contains("response"),
```

```
    # with the z-score function
    z_fun,
    # create new name
    .names = "z_{.col}")) %>%
  mutate(status = "Immigrant",
         RID = as.character(RID))
```

For each native US resident, we include their response and their zipcode profile.

```
us_profile = us_data %>%
  with_groups(RID, filter, row_number() == 1) %>%
   pivot_longer(
    cols = starts_with("q_"),
    names_to = "item",
    values_to = "response",
    values_drop_na = T
  ) %>%
  #merge with location profiles
  left_join(zip_profile_long) %>%
  filter(!is.na(zip_response)) %>%
  # standardize responses within profile
  with_groups(RID, mutate,
              across(contains("response"),
                     z_fun,
                     .names = "z_{.col}")) %>%
  mutate(status = "US Native",
         RID = as.character(RID))
```

2.7 Normative Profile

Profile similarity analyses can be subject to the normativity-desirability confound (NRC) in which correlates of similarity are confounded with having a normative profile, which in turn tends to be a socially-desirable profile (Wood and Furr (2016)). We will control for this confound by including the normative profile as a covariate in subsequent models. We calculate that profile here.

```
normative_profile = immigrants_profile %>%
  full_join(us_profile) %>%
  select(item, z_response) %>%
  filter(!is.na(z_response)) %>%
  group_by(item) %>%
  summarise(normative = mean(z_response))
```

3 Descriptive Statistics

Here I create the descriptives table comparing immigrants to US natives. This becomes S3.

```r
immigrants %>%
  select(RID, sex, age_group, education, parent_edu, region) %>%
  mutate(group = "Immigrants") %>%
  full_join(select(us_data, RID, sex, age_group, education, parent_edu)) %>%
  mutate(group = case_when(
    group == "Immigrants" ~ "Immigrants",
    TRUE ~ "US Natives")) %>%
  select(-RID) %>%
  mutate(sex = str_to_sentence(sex)) %>%
  pivot_longer(cols = -group, values_to = "Variable") %>%
  with_groups(c(group, name, Variable), count) %>%
  with_groups(c(group, name), ~mutate(., p =100*n/sum(n))) %>%
  mutate(estimate = paste0(n, " (", papaja::printnum(p), "\\%)")) %>%
  select(-n, -p) %>%
  pivot_wider(names_from = group, values_from = estimate) %>%
  arrange(name, Variable) %>%
  mutate(
    Variable = ifelse(is.na(Variable), "Missing", Variable),
    Variable = str_remove(Variable, "^[0-9] ")) %>%
  select(-name) %>%
  kable(caption = "Descriptive statistics for both the immigrant sample and sample of US natives.",
        longtable = T,
        #format = "latex",
        align = c("l", "c", "c"),
        escape = FALSE,
        booktabs = TRUE) %>%
  kable_styling(latex_options = c("repeat_header")) %>%
  group_rows("Age", 1, 7) %>%
  group_rows("Education", 8, 16) %>%
  group_rows("Parent Education", 17, 25) %>%
  group_rows("Region", 26, 34) %>%
  group_rows("Sex", 35, 37)
```

Table S3: Descriptive statistics for both the immigrant sample and sample of US natives.

Variable	Immigrants	US Natives
Age		
13-17	7311 (18.65%)	30485 (12.33%)
18-25	16893 (43.10%)	72326 (29.24%)
26-35	7358 (18.77%)	43737 (17.68%)
36-45	3698 (9.43%)	33028 (13.35%)
46-55	2289 (5.84%)	28413 (11.49%)
56-65	1102 (2.81%)	24788 (10.02%)
66+	544 (1.39%)	14555 (5.88%)
Education		
AssociateDegree	1036 (2.64%)	13664 (5.52%)
CollegeDegree	6825 (17.41%)	47202 (19.08%)

Table S3: Descriptive statistics for both the immigrant sample and sample of US natives. *(continued)*

Variable	Immigrants	US Natives
CurrentInUniv	9521 (24.29%)	50366 (20.36%)
GradOrProDegree	6462 (16.49%)	44941 (18.17%)
HSgrad	5622 (14.34%)	20815 (8.42%)
InGradOrProSchool	1673 (4.27%)	9171 (3.71%)
SomeCollege	1837 (4.69%)	24250 (9.80%)
less12yrs	3275 (8.36%)	25316 (10.24%)
Missing	2944 (7.51%)	11607 (4.69%)
Parent Education		
AssociateDegree	701 (1.79%)	5191 (2.10%)
CollegeDegree	5951 (15.18%)	33258 (13.45%)
CurrentInUniv	726 (1.85%)	1114 (0.45%)
GradOrProDegree	7065 (18.03%)	45690 (18.47%)
HSgrad	8428 (21.50%)	61738 (24.96%)
InGradOrProSchool	473 (1.21%)	1903 (0.77%)
SomeCollege	3815 (9.73%)	48118 (19.45%)
less12yrs	5727 (14.61%)	28754 (11.63%)
Missing	6309 (16.10%)	21566 (8.72%)
Region		
Africa	6121 (15.62%)	
East Asia	1323 (3.38%)	
Europe	4721 (12.04%)	
Middle East	1841 (4.70%)	
North America	1529 (3.90%)	
Oceania	633 (1.62%)	
South and Central America	6785 (17.31%)	
South/Southeast Asia	16235 (41.42%)	
Missing	7 (0.02%)	247332 (100.00%)
Sex		
Female	26071 (66.52%)	164989 (66.71%)
Male	12558 (32.04%)	79955 (32.33%)
Other	172 (0.44%)	584 (0.24%)
Missing	394 (1.01%)	1804 (0.73%)

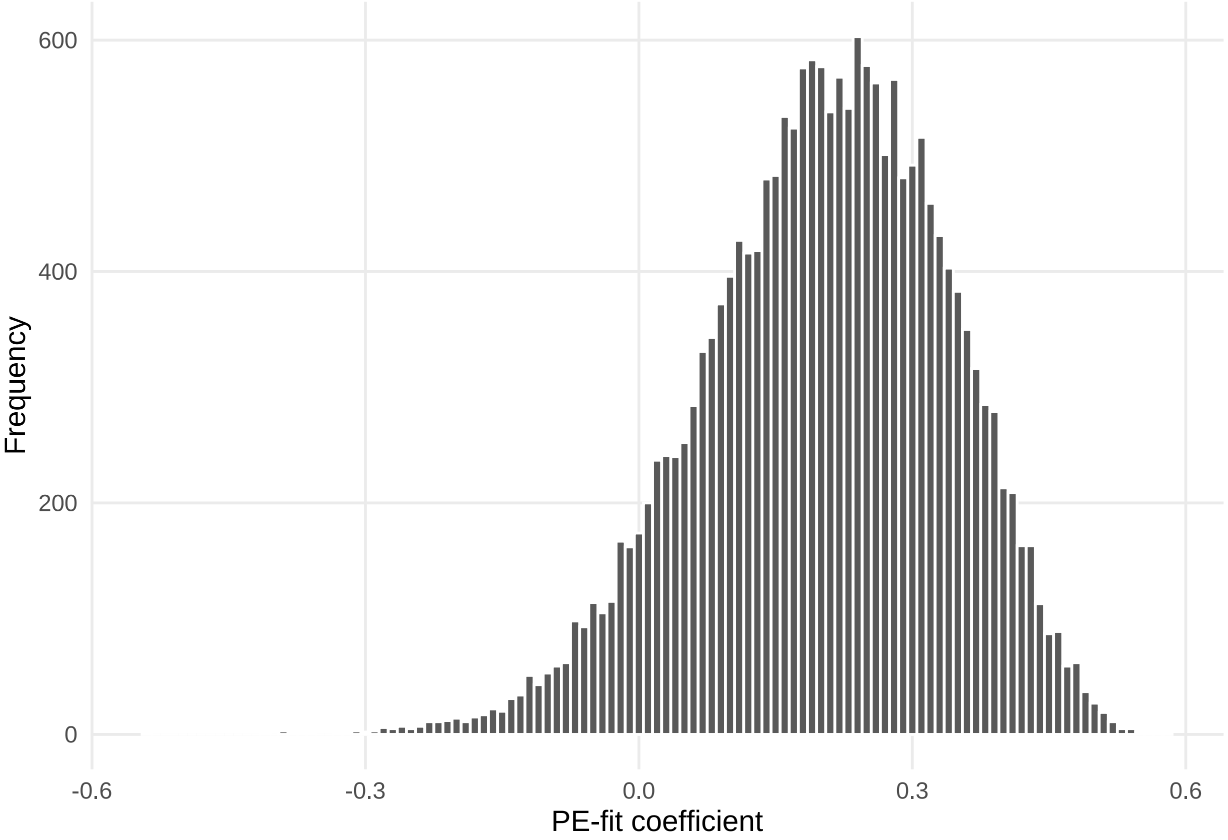
4 Analyses

4.1 RQ1: Do immigrants fit?

To estimate PE-fit, we use a multilevel model with varying slopes (fixed intercepts) and items nested within person. As responses are standardized within participant, the estimated slopes can be interpreted as estimates of correlation coefficients.

The fixed effect (expect fit) can be found in the first model in Table **??**.

```
mod1 = immigrants_profile %>%
  full_join(normative_profile) %>%
  lmer(z_response ~ z_zip_response + normative + (-1 + z_zip_response | ZCTA/RID),
            data = .)
```



We're also interested in knowing whether immigrants fit differently than US natives. To test this question, we fit the same model with all participants and moderate the fit slope by status (Immigrant or US native).

These results are shown in the second model in Table ??.

```
living_in_us = full_join(immigrants_profile, us_profile) %>% full_join(normative_profile)
living_in_us$status = as.factor(living_in_us$status)

mod2 = living_in_us %>%
  lmer(z_response ~ z_zip_response*status + normative + (-1 + z_zip_response | ZCTA/RID),
           data = .)
```

4.2 RQ2: What are the characteristics of individuals that predict fit?

Here we test the demographic and psychological characteristics of US immigrants that may be associated with better PE-fit. For continuous variables, we examine the significance test of the interaction coefficient. For categorical variables, we fit an ANOVA model (Type III sums of squares) with degrees of freedom estimated using the Satterthwaite's method and examine the significance of the interaction in terms of variability explained, rather than linear model coefficient.

Results from these models are in Table S4.

```
mod_age = immigrants_profile %>%
  full_join(normative_profile) %>%
  lmer(z_response ~ z_zip_response*age_group + normative + (-1 + z_zip_response | RID),
           data = .)
```

```r
mod_edu = immigrants_profile %>%
  full_join(normative_profile) %>%
  lmer(z_response ~ z_zip_response*education + normative +  (-1 + z_zip_response | RID),
            data = .)
```

```r
mod_pedu = immigrants_profile %>%
  full_join(normative_profile) %>%
  lmer(z_response ~ z_zip_response*parent_edu + normative + (-1 + z_zip_response | RID),
            data = .)
```

```r
mod_region = immigrants_profile %>%
  full_join(normative_profile) %>%
  lmer(z_response ~ z_zip_response*region + normative + (-1 + z_zip_response | RID),
            data = .)
```

```r
immigrants_profile = immigrants_profile %>%
  mutate(language = factor(language,
                           levels = c("English", "French", "Arabic", "Spanish", "Portuguese", "German",
                           labels = c("English", "French", "Arabic", "Spanish", "Portuguese", "German",

mod_language = immigrants_profile %>%
  full_join(normative_profile) %>%
  lmer(z_response ~ z_zip_response*language + normative + (-1 + z_zip_response | RID),
            data = .)
```

```r
cat_models_df = data.frame(
  predictor = c("Age", "Education", "Parent Education", "World Region", "Official Language")
)

tidy_age      = Anova(mod_age,      test.statistic = "F", type = "III") %>% tidy()
tidy_edu      = Anova(mod_edu,      test.statistic = "F", type = "III") %>% tidy()
tidy_pedu     = Anova(mod_pedu,     test.statistic = "F", type = "III") %>% tidy()
tidy_region   = Anova(mod_region,   test.statistic = "F", type = "III") %>% tidy()
tidy_language = Anova(mod_language, test.statistic = "F", type = "III") %>% tidy()

cat_models_df$sum = list(tidy_age, tidy_edu, tidy_pedu, tidy_region, tidy_language)

save(cat_models_df, file = here("objects/cat_models_df.Rdata"))
```

Table S5: Expected PE-fit byage.

			95 CI	
Group	PE-fit	SE	Lower	Upper
13-17	0.12	0.00	0.11	0.13
18-25	0.18	0.00	0.17	0.18
26-35	0.24	0.00	0.23	0.24
36-45	0.29	0.00	0.28	0.30
46-55	0.32	0.01	0.31	0.33
56-65	0.33	0.01	0.32	0.35
66+	0.33	0.01	0.30	0.35

Table S4: ANOVA summary of categorical individual characteristic models

predictor	PE-fit	Predictor	Normative Profile	Interaction
Age	F(1, 32,823.63) = 1,297.78***	F(6, 1,829,270.29) = 0.01	F(1, 1,842,034.53) = 13,443.25***	F(6, 18,652.78) = 436.75***
Education	F(1, 23,843.07) = 614.73***	F(7, 1,709,412.13) = 0.01	F(1, 1,720,912.74) = 13,690.40***	F(7, 17,967.13) = 214.95***
Parent Education	F(1, 17,818.62) = 438.05***	F(7, 1,562,922.29) = 0.02	F(1, 1,573,184.07) = 12,818.18***	F(7, 16,666.26) = 22.69***
World Region	F(1, 30,706.53) = 2,799.33***	F(7, 1,828,941.99) = 0.03	F(1, 1,841,352.48) = 13,514.75***	F(7, 18,973.21) = 216.55***
Official Language	F(1, 65,536.27) = 6,673.15***	F(7, 1,829,230.50) = 0.02	F(1, 1,841,334.83) = 13,549.07***	F(7, 18,828.65) = 59.89***

Table S6: Test of differences in fit by age.

contrast	estimate	SE	z.ratio	p.value
(13-17) - (18-25)	-0.06	0.00	-17.25	< .001
(13-17) - (26-35)	-0.12	0.00	-28.79	< .001
(13-17) - (36-45)	-0.17	0.00	-34.13	< .001
(13-17) - (46-55)	-0.20	0.01	-35.26	< .001
(13-17) - (56-65)	-0.22	0.01	-27.53	< .001
(13-17) - (66+)	-0.21	0.01	-18.13	< .001
(18-25) - (26-35)	-0.06	0.00	-16.96	< .001
(18-25) - (36-45)	-0.11	0.00	-24.78	< .001
(18-25) - (46-55)	-0.15	0.01	-27.06	< .001
(18-25) - (56-65)	-0.16	0.01	-20.75	< .001
(18-25) - (66+)	-0.15	0.01	-13.17	< .001
(26-35) - (36-45)	-0.05	0.00	-10.50	< .001
(26-35) - (46-55)	-0.09	0.01	-14.97	< .001
(26-35) - (56-65)	-0.10	0.01	-12.45	< .001
(26-35) - (66+)	-0.09	0.01	-7.83	< .001
(36-45) - (46-55)	-0.03	0.01	-5.35	< .001
(36-45) - (56-65)	-0.04	0.01	-5.41	< .001
(36-45) - (66+)	-0.04	0.01	-3.18	.006
(46-55) - (56-65)	-0.01	0.01	-1.18	.710
(46-55) - (66+)	0.00	0.01	-0.24	> .999
(56-65) - (66+)	0.01	0.01	0.56	> .999

Table S7: Expected PE-fit byeducation.

			95 CI	
Group	PE-fit	SE	Lower	Upper
less12yrs	0.11	0.00	0.11	0.12
HSgrad	0.14	0.00	0.13	0.15
CurrentInUniv	0.18	0.00	0.17	0.18
SomeCollege	0.21	0.01	0.20	0.23

Table S7: Expected PE-fit byeducation. *(continued)*

Group	PE-fit	SE	Lower	Upper
AssociateDegree	0.24	0.01	0.22	0.26
CollegeDegree	0.24	0.00	0.23	0.24
InGradOrProSchool	0.23	0.01	0.22	0.24
GradOrProDegree	0.27	0.00	0.26	0.28

Table S8: Test of differences in fit by education.

contrast	estimate	SE	z.ratio	p.value
less12yrs - HSgrad	-0.03	0.01	-4.84	< .001
less12yrs - CurrentInUniv	-0.06	0.01	-12.27	< .001
less12yrs - SomeCollege	-0.10	0.01	-13.27	< .001
less12yrs - AssociateDegree	-0.12	0.01	-13.27	< .001
less12yrs - CollegeDegree	-0.12	0.01	-22.65	< .001
less12yrs - InGradOrProSchool	-0.12	0.01	-16.05	< .001
less12yrs - GradOrProDegree	-0.16	0.01	-29.58	< .001
HSgrad - CurrentInUniv	-0.03	0.00	-8.14	< .001
HSgrad - SomeCollege	-0.07	0.01	-10.34	< .001
HSgrad - AssociateDegree	-0.10	0.01	-10.85	< .001
HSgrad - CollegeDegree	-0.09	0.00	-20.30	< .001
HSgrad - InGradOrProSchool	-0.09	0.01	-13.29	< .001
HSgrad - GradOrProDegree	-0.13	0.00	-28.30	< .001
CurrentInUniv - SomeCollege	-0.04	0.01	-5.69	< .001
CurrentInUniv - AssociateDegree	-0.06	0.01	-7.26	< .001
CurrentInUniv - CollegeDegree	-0.06	0.00	-14.90	< .001
CurrentInUniv - InGradOrProSchool	-0.06	0.01	-8.75	< .001
CurrentInUniv - GradOrProDegree	-0.10	0.00	-24.20	< .001
SomeCollege - AssociateDegree	-0.03	0.01	-2.52	.059
SomeCollege - CollegeDegree	-0.02	0.01	-3.24	.007
SomeCollege - InGradOrProSchool	-0.02	0.01	-2.23	.103
SomeCollege - GradOrProDegree	-0.06	0.01	-8.64	< .001
AssociateDegree - CollegeDegree	0.00	0.01	0.44	> .999
AssociateDegree - InGradOrProSchool	0.01	0.01	0.70	> .999
AssociateDegree - GradOrProDegree	-0.03	0.01	-3.67	.002
CollegeDegree - InGradOrProSchool	0.00	0.01	0.48	> .999
CollegeDegree - GradOrProDegree	-0.04	0.00	-8.40	< .001
InGradOrProSchool - GradOrProDegree	-0.04	0.01	-5.98	< .001

Table S9: Expected PE-fit byparents' education.

			95 CI	
Group	PE-fit	SE	Lower	Upper
AssociateDegree	0.21	0.01	0.19	0.22
CollegeDegree	0.20	0.00	0.19	0.21
CurrentInUniv	0.10	0.01	0.08	0.12
GradOrProDegree	0.22	0.00	0.21	0.22
HSgrad	0.19	0.00	0.18	0.20
InGradOrProSchool	0.17	0.01	0.15	0.20
less12yrs	0.19	0.00	0.18	0.19
SomeCollege	0.21	0.00	0.20	0.22

Table S10: Test of differences in fit by parents' education.

contrast	estimate	SE	z.ratio	p.value
AssociateDegree - CollegeDegree	0.00	0.01	0.37	> .999
AssociateDegree - CurrentInUniv	0.10	0.01	7.35	< .001
AssociateDegree - GradOrProDegree	-0.01	0.01	-1.04	> .999
AssociateDegree - HSgrad	0.01	0.01	1.36	> .999
AssociateDegree - InGradOrProSchool	0.03	0.02	1.95	.564
AssociateDegree - less12yrs	0.02	0.01	1.95	.564
AssociateDegree - SomeCollege	0.00	0.01	-0.40	> .999
CollegeDegree - CurrentInUniv	0.10	0.01	9.44	< .001
CollegeDegree - GradOrProDegree	-0.01	0.00	-3.25	.019
CollegeDegree - HSgrad	0.01	0.00	2.30	.279
CollegeDegree - InGradOrProSchool	0.03	0.01	2.10	.428
CollegeDegree - less12yrs	0.02	0.00	3.39	.012
CollegeDegree - SomeCollege	-0.01	0.01	-1.51	> .999
CurrentInUniv - GradOrProDegree	-0.11	0.01	-10.91	< .001
CurrentInUniv - HSgrad	-0.09	0.01	-8.56	< .001
CurrentInUniv - InGradOrProSchool	-0.07	0.02	-4.51	< .001
CurrentInUniv - less12yrs	-0.08	0.01	-7.79	< .001
CurrentInUniv - SomeCollege	-0.11	0.01	-9.90	< .001
GradOrProDegree - HSgrad	0.02	0.00	5.87	< .001
GradOrProDegree - InGradOrProSchool	0.04	0.01	3.24	.019
GradOrProDegree - less12yrs	0.03	0.00	6.62	< .001
GradOrProDegree - SomeCollege	0.01	0.01	1.24	> .999
HSgrad - InGradOrProSchool	0.02	0.01	1.32	> .999
HSgrad - less12yrs	0.01	0.00	1.38	> .999

Table S10: Test of differences in fit by parents' education. *(continued)*

contrast	estimate	SE	z.ratio	p.value
HSgrad - SomeCollege	-0.02	0.01	-3.55	.007
InGradOrProSchool - less12yrs	-0.01	0.01	-0.81	> .999
InGradOrProSchool - SomeCollege	-0.03	0.01	-2.67	.106
less12yrs - SomeCollege	-0.02	0.01	-4.44	< .001

Table S11: Expected PE-fit byregion.

			95 CI	
Group	PE-fit	SE	Lower	Upper
Africa	0.19	0.00	0.18	0.19
East Asia	0.23	0.01	0.22	0.25
Europe	0.26	0.00	0.25	0.27
Middle East	0.19	0.01	0.18	0.20
North America	0.32	0.01	0.31	0.34
Oceania	0.28	0.01	0.26	0.30
South and Central America	0.25	0.00	0.24	0.26
South/Southeast Asia	0.16	0.00	0.15	0.16

Table S12: Test of differences in fit by world region.

contrast	estimate	SE	z.ratio	p.value
Africa - East Asia	-0.05	0.01	-7.16	< .001
Africa - Europe	-0.07	0.00	-15.37	< .001
Africa - Middle East	-0.01	0.01	-0.96	.338
Africa - North America	-0.14	0.01	-19.69	< .001
Africa - Oceania	-0.09	0.01	-9.07	< .001
Africa - South and Central America	-0.06	0.00	-13.67	< .001
Africa - (South/Southeast Asia)	0.03	0.00	8.38	< .001
East Asia - Europe	-0.03	0.01	-3.82	.001
East Asia - Middle East	0.04	0.01	5.16	< .001
East Asia - North America	-0.09	0.01	-10.50	< .001
East Asia - Oceania	-0.04	0.01	-3.91	.001
East Asia - South and Central America	-0.01	0.01	-2.13	.099
East Asia - (South/Southeast Asia)	0.08	0.01	12.81	< .001
Europe - Middle East	0.07	0.01	10.26	< .001
Europe - North America	-0.06	0.01	-8.82	< .001
Europe - Oceania	-0.02	0.01	-1.72	.170
Europe - South and Central America	0.01	0.00	2.49	.052

Table S12: Test of differences in fit by world region. *(continued)*

contrast	estimate	SE	z.ratio	p.value
Europe - (South/Southeast Asia)	0.11	0.00	25.51	< .001
Middle East - North America	-0.13	0.01	-15.77	< .001
Middle East - Oceania	-0.09	0.01	-7.73	< .001
Middle East - South and Central America	-0.06	0.01	-8.71	< .001
Middle East - (South/Southeast Asia)	0.04	0.01	6.40	< .001
North America - Oceania	0.05	0.01	4.00	.001
North America - South and Central America	0.08	0.01	10.77	< .001
North America - (South/Southeast Asia)	0.17	0.01	25.89	< .001
Oceania - South and Central America	0.03	0.01	2.94	.017
Oceania - (South/Southeast Asia)	0.12	0.01	12.58	< .001
South and Central America - (South/Southeast Asia)	0.09	0.00	24.57	< .001

Table S13: Expected PE-fit bylanguage.

			95 CI	
Group	PE-fit	SE	Lower	Upper
English	0.19	0.00	0.19	0.19
French	0.28	0.01	0.26	0.30
Arabic	0.16	0.01	0.15	0.17
Spanish	0.24	0.00	0.23	0.25
Portuguese	0.25	0.01	0.23	0.27
German	0.30	0.01	0.28	0.32
Russian	0.23	0.02	0.20	0.26
Other	0.22	0.00	0.21	0.22

Table S14: Test of differences in fit by official language.

contrast	estimate	SE	z.ratio	p.value
English - French	-0.09	0.01	-8.64	< .001
English - Arabic	0.03	0.01	6.03	< .001
English - Spanish	-0.05	0.00	-10.68	< .001
English - Portuguese	-0.06	0.01	-6.52	< .001
English - German	-0.11	0.01	-11.39	< .001
English - Russian	-0.04	0.02	-2.74	.049
English - Other	-0.03	0.00	-7.88	< .001
French - Arabic	0.13	0.01	10.68	< .001

Table S14: Test of differences in fit by official language. *(continued)*

contrast	estimate	SE	z.ratio	p.value
French - Spanish	0.04	0.01	3.69	.002
French - Portuguese	0.03	0.01	2.32	.123
French - German	-0.02	0.01	-1.50	.668
French - Russian	0.05	0.02	2.55	.076
French - Other	0.07	0.01	6.03	< .001
Arabic - Spanish	-0.08	0.01	-12.14	< .001
Arabic - Portuguese	-0.09	0.01	-8.94	< .001
Arabic - German	-0.15	0.01	-13.18	< .001
Arabic - Russian	-0.08	0.02	-4.65	< .001
Arabic - Other	-0.06	0.01	-9.90	< .001
Spanish - Portuguese	-0.01	0.01	-0.99	> .999
Spanish - German	-0.06	0.01	-5.92	< .001
Spanish - Russian	0.01	0.02	0.39	> .999
Spanish - Other	0.02	0.01	4.55	< .001
Portuguese - German	-0.05	0.01	-4.02	.001
Portuguese - Russian	0.02	0.02	0.89	> .999
Portuguese - Other	0.03	0.01	3.55	.004
German - Russian	0.07	0.02	3.75	.002
German - Other	0.09	0.01	8.53	< .001
Russian - Other	0.02	0.02	1.07	> .999

Next, we examine differences by country. For these analyses, we use only those countries that have at least 50 participants – this is meant to limit sampling error in the estimation of average fit.

The countries with the highest and lowest fit are shown in Figure S1.

```
mod_country = immigrants_profile %>%
  with_groups(country,
              ~mutate(., n = n())) %>%
  filter(n >= 50) %>%
  full_join(normative_profile) %>%
  lmer(z_response ~ z_zip_response*country + normative + (-1 + z_zip_response | RID),
          data = .)

country_trends = emtrends(mod_country,
                      ~country,
                      var = "z_zip_response",
                      pbkrtest.limit = 1848641,
                      adjust = "holm")



save(mod_country, country_trends, file = here("objects/mod-country.Rdata"))
```

A final set of models in this section (immigrant characteristics) tests whether some personality traits are more strongly associated with fit than others. We examine both broad (5-factor) and narrow (26-factor) traits. We test both linear and polynomial terms, using two separate models. To assist with interpretation, we center trait scores prior to estimation.

Results can be found in Table **??**.

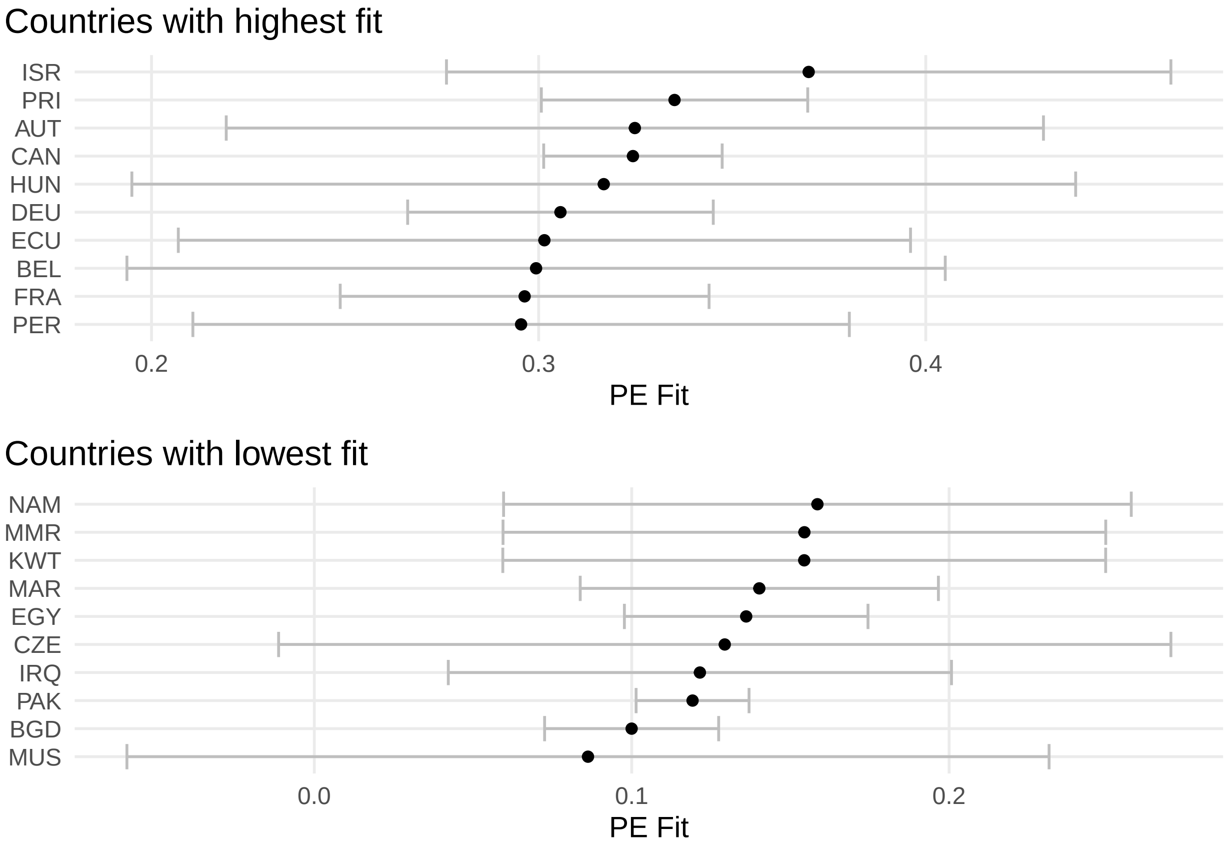


Figure S1: Countries with immigrants who have the the highest and lowest fit.

```r
b5_names = immigrants_profile %>%
  select(starts_with("SPI_")) %>%
  names()

spi_names = immigrants_profile %>%
  select(starts_with("SE_")) %>%
  names() %>%
  str_remove("SE_")

spi_names = spi_names[!spi_names == "WellBeing"]
```

```r
for(i in c(b5_names, spi_names)){

  mod_obj = immigrants_profile %>%
    with_groups(RID, ~filter(., row_number() == 1)) %>%
    # center personality
    mutate(
      across(all_of(i),
             ~. - mean(., na.rm = T),
             .names = "trait_score")
    ) %>%
    select(RID, trait_score) %>%
    full_join(immigrants_profile) %>%
    select(RID, ZCTA, z_response, z_zip_response, trait_score) %>%
    nest() %>%
    mutate(trait = i) %>%
    mutate(model1 = purrr::map(data,
                               ~lmer(z_response ~
                                       z_zip_response*trait_score +
                                       (-1 + z_zip_response | RID),
                                     data = .)),
           model2 = purrr::map(data,
                               ~lmer(z_response ~
                                       z_zip_response*trait_score +
                                       z_zip_response*I(trait_score^2) +
                                       (-1 + z_zip_response | RID),
                                     data = .))) %>%
    select(-data)

  save(mod_obj, file = here("objects/models-personality/", str_c(i, ".Rdata")))

  place = which(c(b5_names, spi_names) == i)
  print(str_c(i, " complete (", place, "/31)"))
}
```

```r
# load models
personality_mods_files = list.files(here("objects/models-personality"))

# load all personality models and bind to single data frame
for(i in personality_mods_files){

  load(file = here("objects/models-personality/", i))

```

```
  if(i == personality_mods_files[1]){

      personality_mods = mod_obj

      quad_slopes = emmeans::emtrends(mod_obj$model2[[1]],
                                    var = "z_zip_response",
                                    at = list(
                                      trait_score = seq(-2,2,
                                                        by = .1))
                                    ) %>%
      as.data.frame() %>%
      mutate(trait = mod_obj$trait[[1]])

      } else {

        personality_mods = rbind(personality_mods, mod_obj)

        slope_est = emmeans::emtrends(mod_obj$model2[[1]],
                                    var = "z_zip_response",
                                    at = list(
                                      trait_score = seq(-2,2,
                                                        by = .1))
                                    ) %>%
          as.data.frame() %>%
          mutate(trait = mod_obj$trait[[1]])

        quad_slopes = rbind(quad_slopes, slope_est)
  }
}

save(quad_slopes, file = here("objects/quad_slopes.Rdata"))

personality_table = personality_mods %>%
  mutate(model = purrr::map(model2, tidy, conf.int = T)) %>%
  select(trait, model) %>%
  unnest(cols = c(model))

save(personality_table,
     file = here("objects/personality-table-data.Rdata"))
```

```
load(here("objects/personality-table-data.Rdata"))

personality_table = personality_table %>%
  filter(str_detect(term, ":")) %>%
  mutate(trait = str_remove(trait, "SPI_")) %>%
  mutate(p.value = map_chr(p.value, papaja::printp),
         across(where(is.numeric), papaja::printnum),
         level = case_when(
           trait %in% c("Agree", "Consc",
                        "Extra", "Neuro",
                        "Open")            ~ "Broad",
           TRUE                            ~ "Narrow"
         ),
```

```
         trait = case_when(
           trait == "Agree" ~ "Agreeableness",
           trait == "Consc" ~ "Conscientiousness",
           trait == "Extra" ~ "Extraversion",
           trait == "Neuro" ~ "Neuroticism",
           trait == "Open" ~ "Openness",
           TRUE ~ trait
         ),
         trait = str_replace(trait, "([a-z])([A-Z])", "\\1 \\2"),
         ci = paste0("[", conf.low, ", ", conf.high,"]"),
         term = ifelse(str_detect(term, "2"), "Quadratic", "Linear")) %>%
  select(level, trait, term, estimate, ci, p.value) %>%
  pivot_longer(cols = c(estimate, ci, p.value),
               names_to = "stat", values_to = "stat_value") %>%
  unite(term, term, stat) %>%
  pivot_wider(names_from = term, values_from = stat_value) %>%
  arrange(level) %>%
  select(-level) %>%
  kable(booktabs = T,
        escape = F,
        format = "latex",
        col.names = c(" ", "Est", "95\\% CI", "p", "Est", "95\\% CI", "p"),
        caption = "Association between personality traits and PE-fit among immigrants") %>%
  kable_styling() %>%
  add_header_above(c(" ", "Linear" = 3, "Quadratic" = 3)) %>%
  group_rows("Broad traits", 1, 5) %>%
  group_rows("Narrow traits", 6, 31)

save(personality_table,
     file = here("objects/personality-table.Rdata"))
```

```
load(here("objects/quad_slopes.Rdata"))
# which lines are mostly negative, not positive

neg_slopes = quad_slopes %>%
  filter(trait_score == 2) %>%
  filter(z_zip_response.trend < .345)

quad_slopes = quad_slopes %>%
  mutate(
    group = case_when(
      trait %in% neg_slopes$trait ~ 1,
      TRUE ~ 2
    )
  )

## ends of lines
data_ends <- quad_slopes %>%
  filter(trait_score == 2)

quad_slopes %>%
  ggplot(aes(x = trait_score, y = z_zip_response.trend, color = trait)) +
  geom_line() +
```

```
  geom_text_repel(
    aes(label = trait),
    data = data_ends,
    color = "black",
    nudge_x = 1,
    direction = "y",
    vjust = 1.5,
    hjust = 1,
    inherit.aes = T
    ) +
  scale_x_continuous(limits = c(-2, 3)) +
  guides(color = "none") +
  facet_wrap(~group) +
  labs(
    x = "Trait score",
    y = "Estimated PE fit"
  ) +
  theme_clean()
```

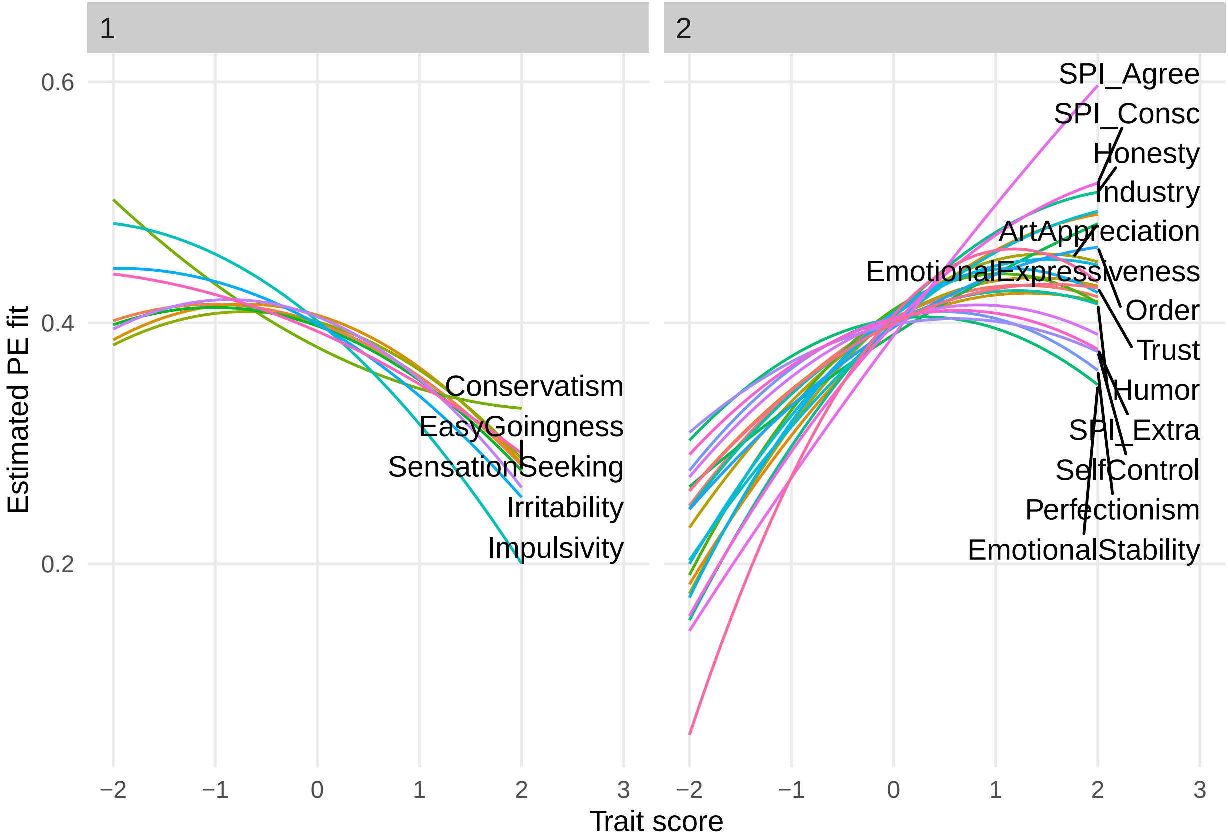


Figure S2: Estimated PE-fit by trait from quadratic models.

```
ggsave(filename = here("figures/quad_slopes.pdf"), width = 10*2, height = 4*2)
```

Table S15: Association between personality traits and PE-fit among immigrants

	Linear			Quadratic		
	Est	95% CI	p	Est	95% CI	p
Broad traits						
Agreeableness	0.11	[0.11, 0.12]	< .001	0.00	[-0.01, 0.00]	< .001
Conscientiousness	0.09	[0.09, 0.09]	< .001	-0.02	[-0.02, -0.01]	< .001
Extraversion	0.02	[0.02, 0.02]	< .001	-0.02	[-0.02, -0.02]	< .001
Neuroticism	-0.04	[-0.04, -0.03]	< .001	-0.01	[-0.01, 0.00]	< .001
Openness	0.09	[0.09, 0.10]	< .001	-0.04	[-0.04, -0.04]	< .001
Narrow traits						
Adaptability	0.04	[0.04, 0.05]	< .001	-0.02	[-0.02, -0.02]	< .001
Anxiety	-0.03	[-0.03, -0.03]	< .001	-0.01	[-0.02, -0.01]	< .001
Art Appreciation	0.08	[0.07, 0.08]	< .001	-0.02	[-0.02, -0.01]	< .001
Attention Seeking	-0.03	[-0.03, -0.02]	< .001	-0.02	[-0.02, -0.02]	< .001
Authoritarianism	0.04	[0.04, 0.04]	< .001	-0.01	[-0.02, -0.01]	< .001
Charisma	0.05	[0.05, 0.05]	< .001	-0.02	[-0.02, -0.02]	< .001
Compassion	0.07	[0.07, 0.07]	< .001	-0.02	[-0.03, -0.02]	< .001
Conformity	-0.02	[-0.03, -0.02]	< .001	-0.02	[-0.02, -0.01]	< .001
Conservatism	-0.04	[-0.05, -0.04]	< .001	0.01	[0.01, 0.01]	< .001
Creativity	0.06	[0.05, 0.06]	< .001	-0.03	[-0.03, -0.02]	< .001
Easy Goingness	-0.03	[-0.03, -0.03]	< .001	-0.01	[-0.02, -0.01]	< .001
Emotional Expressiveness	0.05	[0.05, 0.06]	< .001	0.00	[-0.01, 0.00]	< .001
Emotional Stability	0.01	[0.01, 0.01]	< .001	-0.02	[-0.02, -0.02]	< .001
Honesty	0.09	[0.09, 0.09]	< .001	-0.02	[-0.02, -0.02]	< .001
Humor	0.04	[0.04, 0.05]	< .001	-0.02	[-0.02, -0.02]	< .001
Impulsivity	-0.07	[-0.07, -0.07]	< .001	-0.02	[-0.02, -0.01]	< .001
Industry	0.07	[0.07, 0.07]	< .001	-0.01	[-0.01, -0.01]	< .001
Intellect	0.06	[0.06, 0.06]	< .001	-0.02	[-0.02, -0.02]	< .001
Introspection	0.06	[0.06, 0.07]	< .001	-0.03	[-0.03, -0.03]	< .001
Irritability	-0.05	[-0.05, -0.05]	< .001	-0.01	[-0.01, -0.01]	< .001
Order	0.05	[0.05, 0.06]	< .001	-0.01	[-0.01, -0.01]	< .001
Perfectionism	0.02	[0.02, 0.02]	< .001	-0.02	[-0.02, -0.02]	< .001
Self Control	0.02	[0.01, 0.02]	< .001	-0.01	[-0.02, -0.01]	< .001
Sensation Seeking	-0.03	[-0.04, -0.03]	< .001	-0.02	[-0.02, -0.02]	< .001
Sociability	0.03	[0.03, 0.03]	< .001	-0.02	[-0.02, -0.02]	< .001
Trust	0.04	[0.04, 0.04]	< .001	-0.01	[-0.02, -0.01]	< .001

```
person_pefit_figure = personality_mods %>%
  mutate(model = purrr::map(model1, tidy, conf.int = T)) %>%
  select(trait, model) %>%
  unnest(cols = c(model)) %>%
  filter(str_detect(term, ":")) %>%
  mutate(trait = str_remove(trait, "SPI_")) %>%
  mutate(level = case_when(
           trait %in% c("Agree", "Consc", "Extra", "Neuro", "Open") ~ "Broad",
           TRUE ~ "Narrow"),
         color = case_when(
           trait %in% c("Extra", "Sociability", "Humor",
                        "AttentionSeeking", "Charisma",
                        "SensationSeeking", "EmotionalExpressiveness") ~ "E",
           trait %in% c("Agree", "Honesty", "Trust", "Compassion") ~ "A",
           trait %in% c("Consc", "Industry", "Impulsivity",
                        "EasyGoingness", "Order", "Authoritarianism",
                        "Perfectionism", "Conservatism") ~ "C",
           trait %in% c("Neuro","Adaptability", "EmotionalStability",
                        "Anxiety", "SelfControl", "Irritability") ~ "N",
           trait %in% c("Open" ,"Intellect", "ArtAppreciation", "Conformity",
                        "Introspection", "Creativity") ~ "O"),
         trait = case_when(
           trait == "Agree" ~ "Agreeableness",
           trait == "Consc" ~ "Conscientiousness",
           trait == "Extra" ~ "Extraversion",
           trait == "Neuro" ~ "Neuroticism",
           trait == "Open" ~ "Openness",
           TRUE ~ trait
         ),
         trait = str_replace(trait, "([a-z])([A-Z])", "\\1 \\2"),
         sig = case_when(
           p.value < .05 & estimate > 0 ~ "Pos",
           p.value < .05 & estimate < 0 ~ "Neg",
           TRUE ~ "NOT")) %>%
  ggplot(aes(x = reorder(trait,estimate), y = estimate)) +
  geom_bar(aes(fill = color),
           stat = "identity") +
  geom_hline(aes(yintercept = 0)) +
  geom_errorbar(
    aes(ymin = conf.low,
        ymax = conf.high,
        alpha = sig),
    width = .5) +
  scale_fill_viridis_d(option = "plasma", end = 0.8) +
  scale_alpha_manual(values = c(1, .3, 1)) +
  labs(x = NULL, y = "Effect on PE-fit") +
  guides(fill = "none", alpha = "none") +
  coord_flip() +
  facet_grid(level ~., scales = "free_y", space = "free") +
  theme_clean()

save(person_pefit_figure,
     file = here("objects/personality-figure.Rdata"))
```

4.3 RQ3: What are the characteristics of communities that predict fit?

Here we test the features of US communities that may be associated with better PE-fit among their immigrants. For continuous variables, we examine the significance test of the interaction coefficient. For categorical variables, we fit an ANOVA model (Type III sums of squares) with degrees of freedom estimated using the Satterthwaite's method and examine the significance of the interaction in terms of variability explained, rather than linear model coefficient.

Results from these models are in Table S17.

A note that proportion values are divided by 100, to aid with model convergence. We also center all continuous predictors, to facilitate comparison of linear and quadratic models.

```r
immigrants_profile = immigrants_profile %>%
  mutate(prop_college = prop_college/100,
         prop_foreign = prop_foreign/100,
         prop_originContinent = prop_originContinent/100,
         prop_ancestry = prop_ancestry/100,
         match_originCountry = match_originCountry/100)

immigrants_profile = immigrants_profile %>%
  with_groups(RID, ~filter(., row_number() == 1)) %>%
  select(RID, prop_college, prop_foreign,
         prop_originContinent, prop_ancestry, match_originCountry) %>%
  mutate(
    across(
      c(-RID),
      ~. - mean(., na.rm=T), .names = "{.col}_c")) %>%
  full_join(immigrants_profile)
```

```r
mod_college1 = immigrants_profile %>%
  full_join(normative_profile) %>%
  lmer(z_response ~
         z_zip_response*prop_college +
         normative +
         (-1 + z_zip_response | RID),
            data = .)

mod_college2 = immigrants_profile %>%
  full_join(normative_profile) %>%
  lmer(z_response ~
         z_zip_response*prop_college +
         z_zip_response*I(prop_college^2) +
         normative +
         (-1 + z_zip_response | RID),
            data = .)
```

```r
load(here("objects/college.Rdata"))
```

```r
mod_regionZ = immigrants_profile %>%
  full_join(normative_profile) %>%
  lmer(z_response ~
         z_zip_response*zcta_region +
```

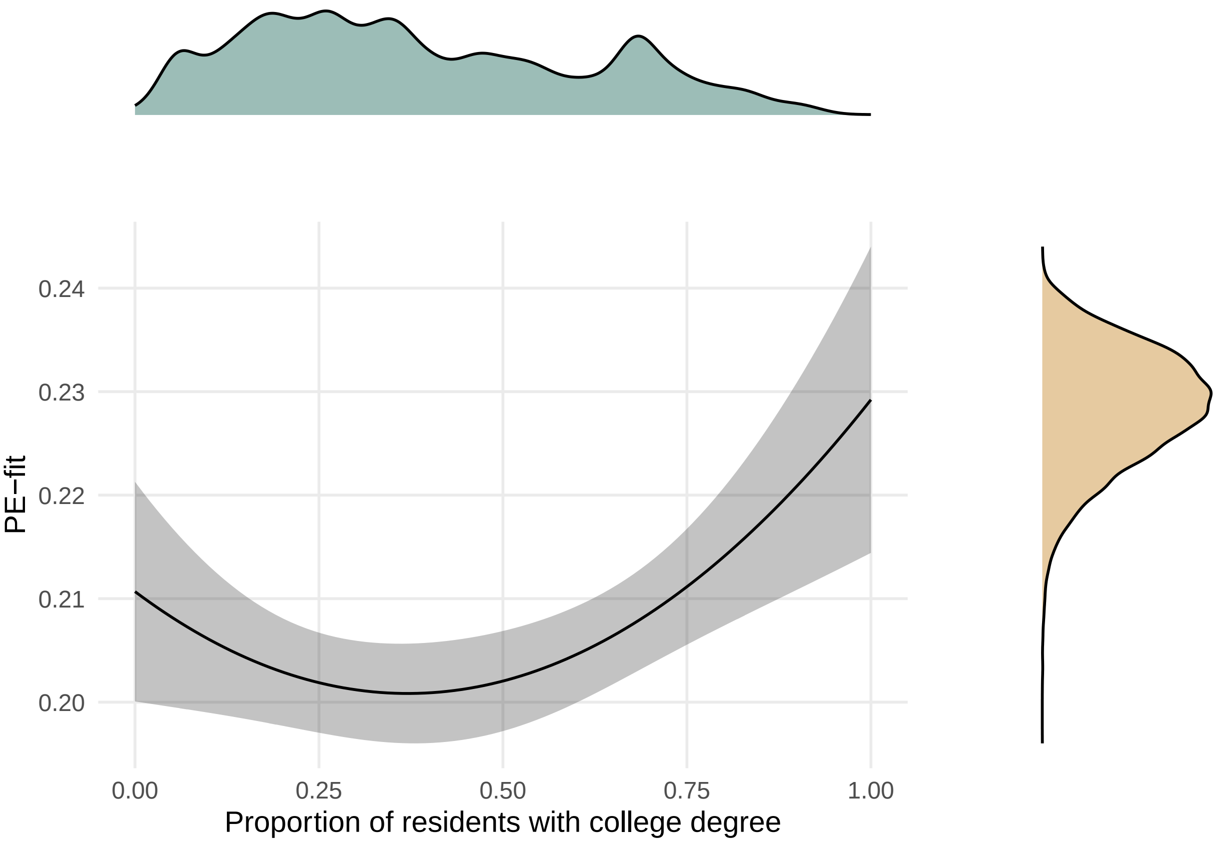


Figure S3: Relationship of college degrees to PE-fit of immigrants. Figure is based on the quadratic model fit.

```
         normative +
         (-1 + z_zip_response | RID),
            data = .)

sum_mod_regionZ = Anova(mod_regionZ, type = "III",
                        test.statistic = "F")
save(mod_regionZ, sum_mod_regionZ,
     file = here("objects/regionZ.Rdata"))
```

Table S16: Test of differences in fit by US region.

contrast	estimate	SE	z.ratio	p.value
(Mid-Atlantic) - Midwest	-0.05	0.00	-9.28	< .001
(Mid-Atlantic) - New England	-0.07	0.01	-7.01	< .001
(Mid-Atlantic) - Pacific Coastal	-0.03	0.00	-8.58	< .001
(Mid-Atlantic) - Rocky Mountains	-0.01	0.01	-1.52	.383
(Mid-Atlantic) - Southeast	-0.04	0.00	-9.36	< .001
(Mid-Atlantic) - Southwest	-0.01	0.00	-2.85	.035
Midwest - New England	-0.02	0.01	-2.03	.171
Midwest - Pacific Coastal	0.02	0.01	3.46	.005
Midwest - Rocky Mountains	0.04	0.01	4.71	< .001
Midwest - Southeast	0.01	0.01	1.11	.538
Midwest - Southwest	0.03	0.01	5.42	< .001
New England - Pacific Coastal	0.04	0.01	3.98	.001
New England - Rocky Mountains	0.06	0.01	5.08	< .001
New England - Southeast	0.03	0.01	2.72	.039
New England - Southwest	0.05	0.01	5.27	< .001
Pacific Coastal - Rocky Mountains	0.02	0.01	2.78	.038
Pacific Coastal - Southeast	-0.01	0.00	-2.58	.050
Pacific Coastal - Southwest	0.02	0.00	3.21	.012
Rocky Mountains - Southeast	-0.03	0.01	-4.13	< .001
Rocky Mountains - Southwest	0.00	0.01	-0.44	.656
Southeast - Southwest	0.03	0.01	4.85	< .001

```
mod_foreign1 = immigrants_profile %>%
  full_join(normative_profile) %>%
  lmer(z_response ~
         z_zip_response*prop_foreign +
         normative +
         (-1 + z_zip_response | RID),
            data = .)

mod_foreign2 = immigrants_profile %>%
  full_join(normative_profile) %>%
  lmer(z_response ~
         z_zip_response*prop_foreign +
```

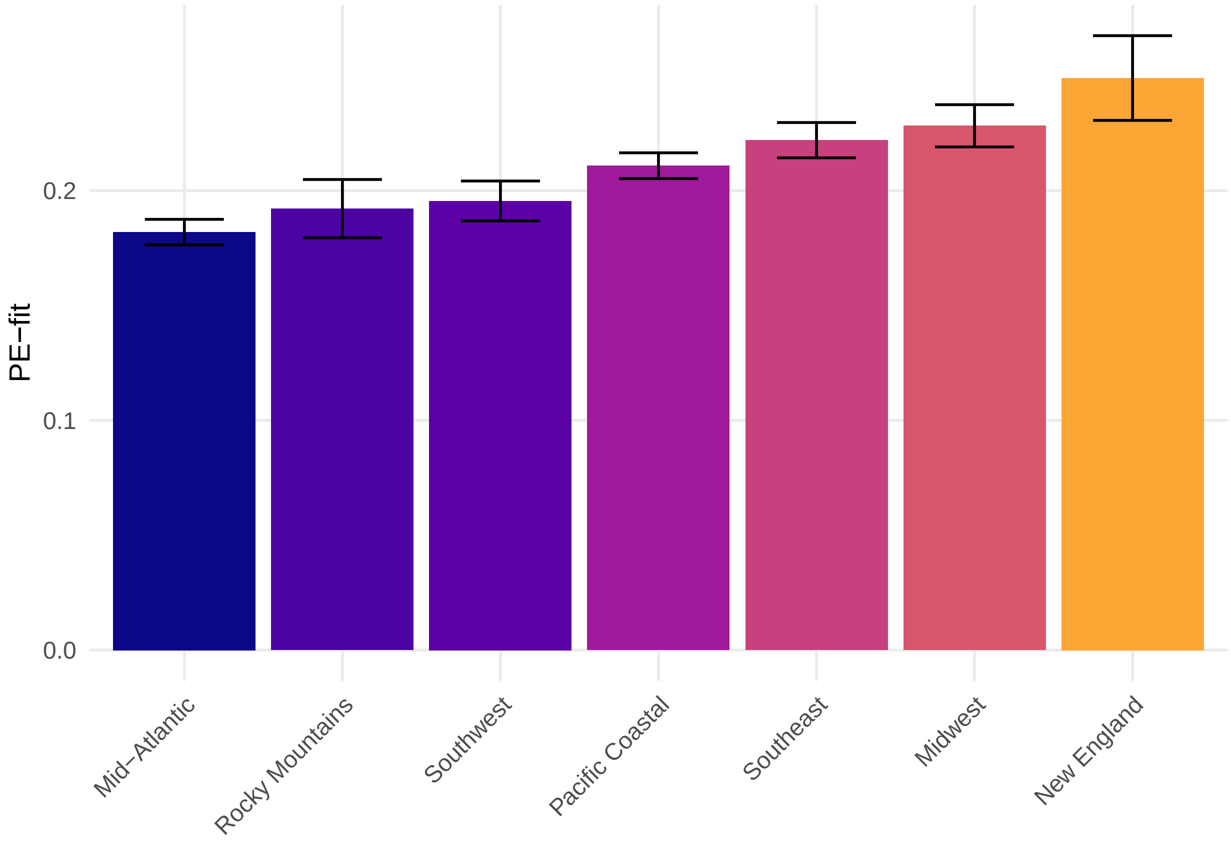


Figure S4: Estimated PE-fit by continent of origin.

```
         z_zip_response*I(prop_foreign^2) +
         normative +
         (-1 + z_zip_response | RID),
            data = .)

save(mod_foreign1, mod_foreign2, file = here("objects/foreign.Rdata"))
```

```
load(here("objects/foreign.Rdata"))
```

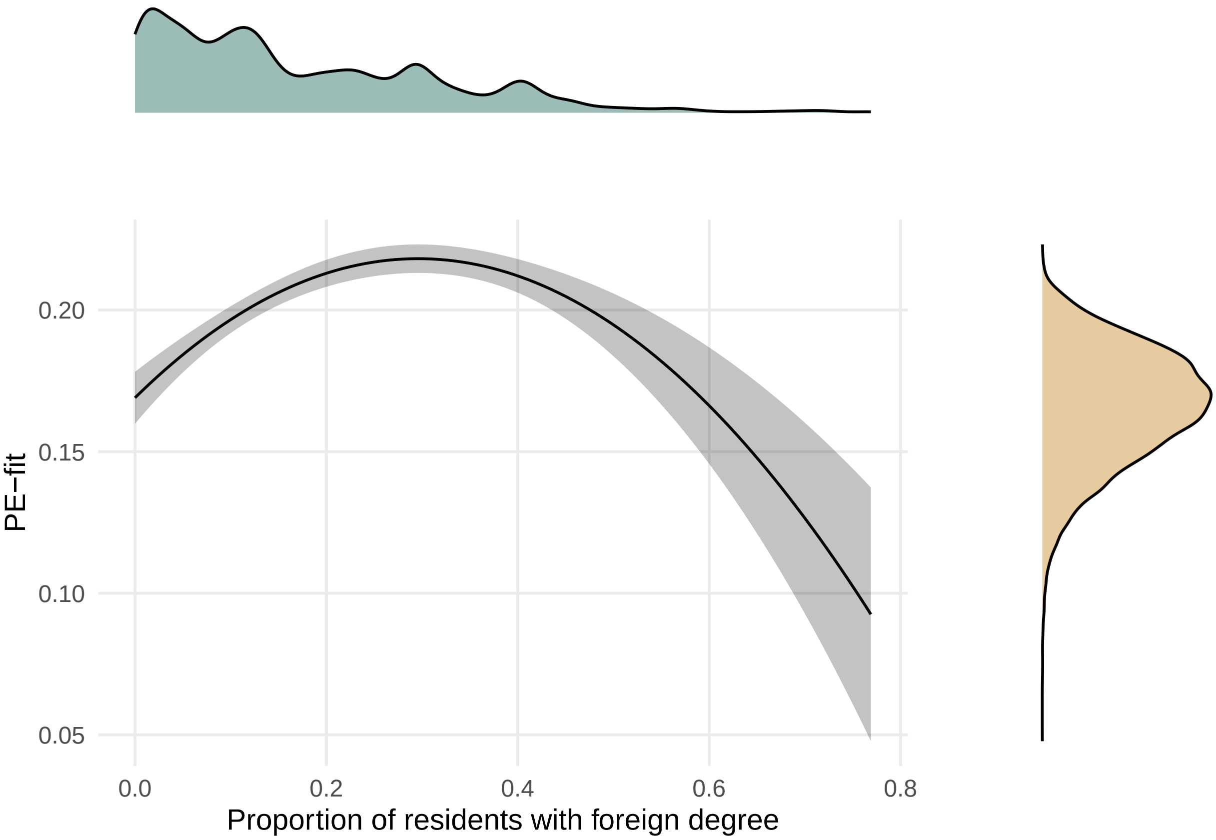


Figure S5: Relationship of foreign-born residents to PE-fit of immigrants. Figure is based on the quadratic model fit.

```
mod_originContinent1 = immigrants_profile %>%
  full_join(normative_profile) %>%
  lmer(z_response ~
         z_zip_response*prop_originContinent +
         normative +
         (-1 + z_zip_response | RID),
            data = .)

mod_originContinent2 = immigrants_profile %>%
  full_join(normative_profile) %>%
  lmer(z_response ~
         z_zip_response*prop_originContinent +
         z_zip_response*I(prop_originContinent^2) +
```

```
         normative +
         (-1 + z_zip_response | RID),
            data = .)

save(mod_originContinent1, mod_originContinent2,
     file = here("objects/originContinent.Rdata"))
```

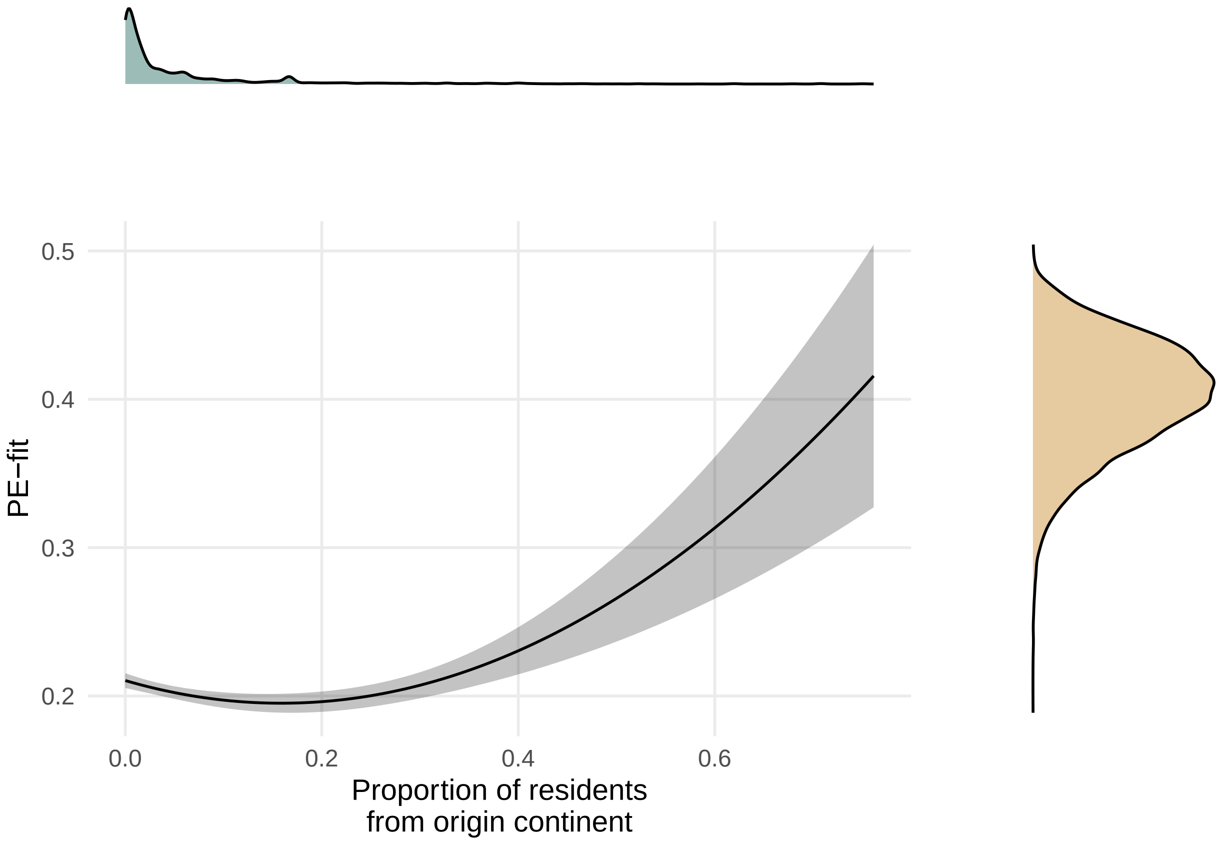


Figure S6: Relationship of immigrants who match continent-of-origin to PE-fit of immigrants. Figure is based on the quadratic model fit.

```
mod_originCountry1 = immigrants_profile %>%
  full_join(normative_profile) %>%
  lmer(z_response ~
         z_zip_response*match_originCountry +
         normative +
         (-1 + z_zip_response | RID),
            data = .)

mod_originCountry2 = immigrants_profile %>%
  full_join(normative_profile) %>%
  lmer(z_response ~
         z_zip_response*match_originCountry +
         z_zip_response*I(match_originCountry^2) +
         normative +
         (-1 + z_zip_response | RID),
```

```
           data = .)

save(mod_originCountry1, mod_originCountry2, file = here("objects/originCountry.Rdata"))
```

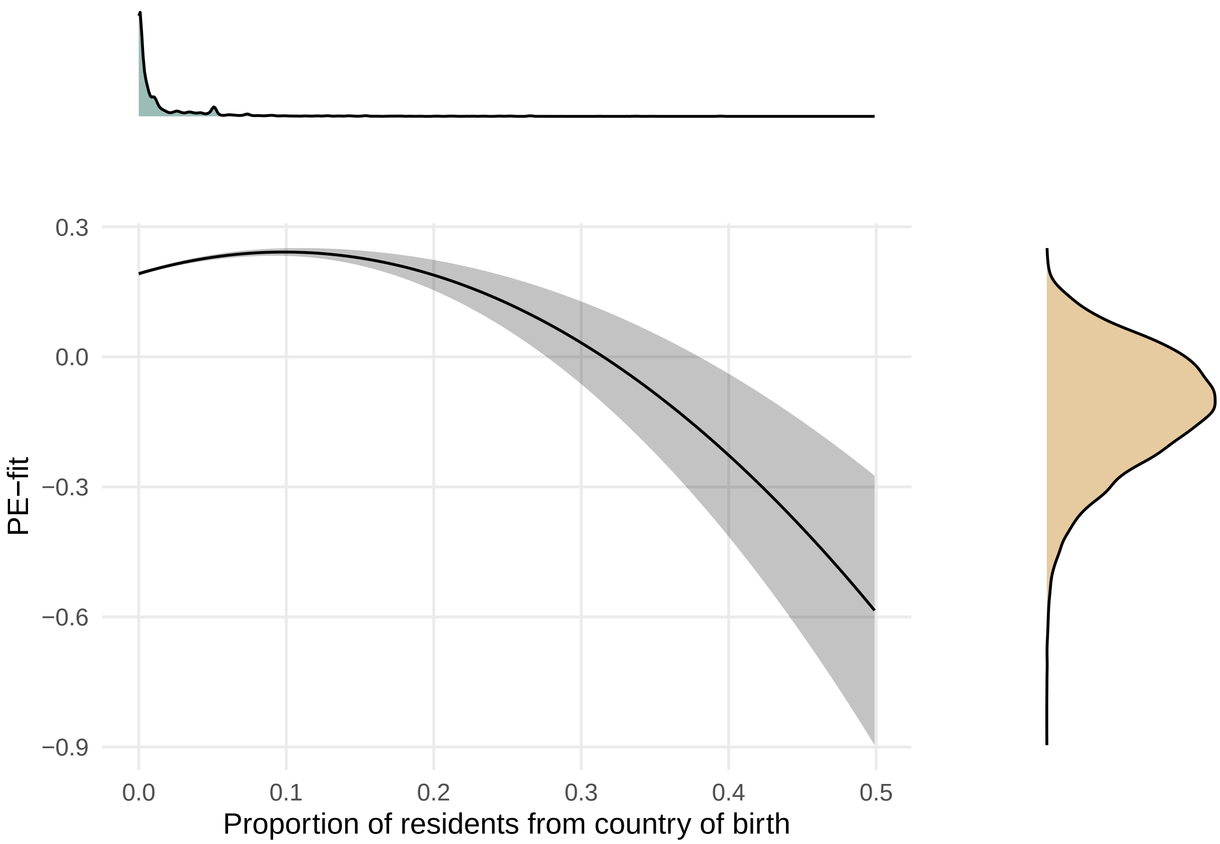


Figure S7: Relationship of immigrants who match country of origin to PE-fit of immigrants. Figure is based on the quadratic model fit.

```
mod_ancestry1 = immigrants_profile %>%
  full_join(normative_profile) %>%
  lmer(z_response ~
         z_zip_response*prop_ancestry +
         (-1 + z_zip_response | RID),
           data = .)

mod_ancestry2 = immigrants_profile %>%
  full_join(normative_profile) %>%
  lmer(z_response ~
         z_zip_response*prop_ancestry +
         z_zip_response*I(prop_ancestry^2) +
         (-1 + z_zip_response | RID),
           data = .)

save(mod_ancestry1, mod_ancestry2, file = here("objects/ancestry.Rdata"))
```

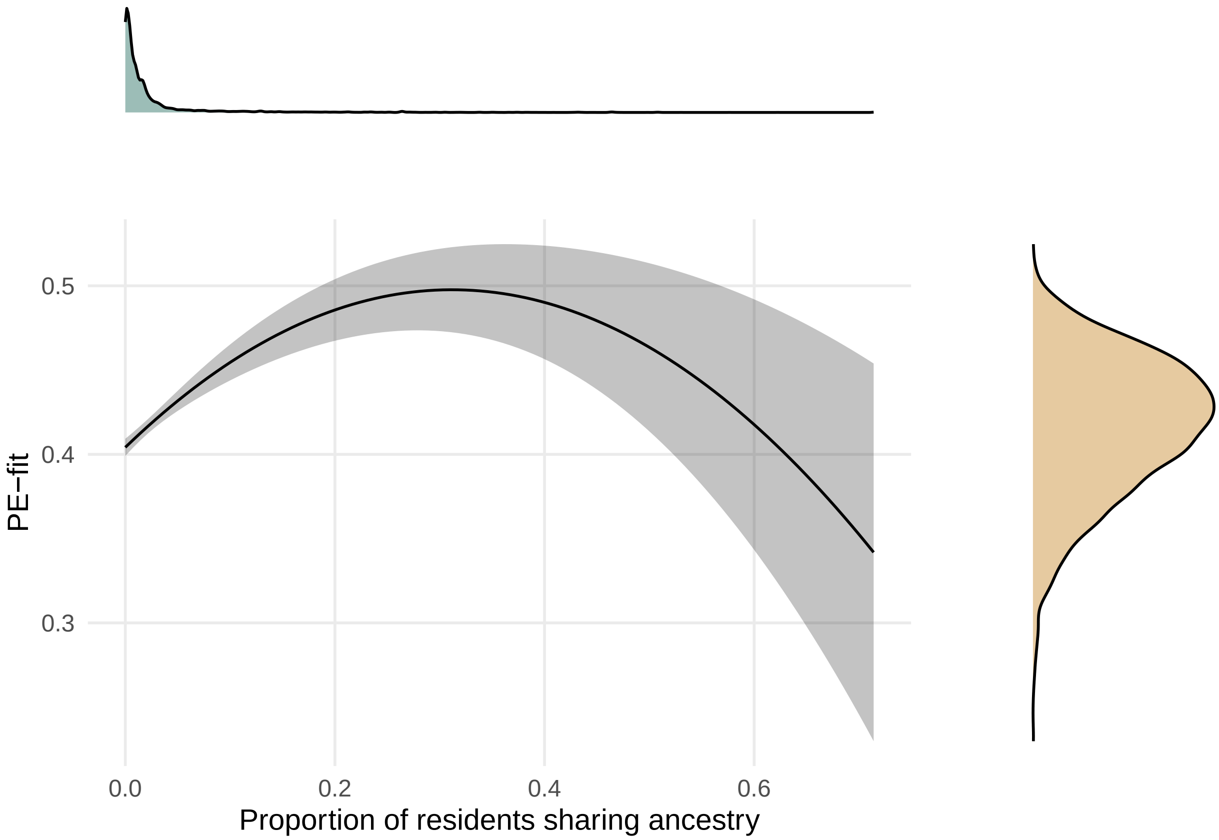


Figure S8: Relationship of immigrants who match continent-of-ancestry to PE-fit of immigrants. Figure is based on the quadratic model fit.

Table S17: Association between ZCTA characteristics and PE-fit among immigrants

		Linear			Quadratic		
	Term	Est	95% CI	p	Est	95% CI	p
Proportion foreign born	b	0.06	[0.04, 0.08]	< .001	0.33	[0.25, 0.41]	< .001
Proportion foreign born	b^2				-0.56	[-0.73, -0.40]	< .001
Proportion from country of birth	b	0.45	[0.35, 0.55]	< .001	1.02	[0.82, 1.21]	< .001
Proportion from country of birth	b^2				-5.16	[-6.73, -3.58]	< .001
Proportion from origin continent	b	-0.01	[-0.04, 0.03]	.743	-0.19	[-0.27, -0.11]	< .001
Proportion from origin continent	b^2				0.61	[0.36, 0.85]	< .001
Proportion match ancestry	b	0.26	[0.17, 0.34]	< .001	0.60	[0.44, 0.76]	< .001
Proportion match ancestry	b^2				-0.96	[-1.33, -0.60]	< .001
Proportion with college degree	b	0.01	[0.00, 0.02]	.043	-0.05	[-0.10, 0.00]	.033
Proportion with college degree	b^2				0.07	[0.02, 0.12]	.007

```
zcta_locations = geocode_zip(immigrants$ZCTA) %>%
  rename(
    ZCTA = zipcode,
    ZCTAlat = lat,
    ZCTAlon = lng) %>%
  mutate(ZCTA = as.numeric(ZCTA))

immigrants = full_join(immigrants, zcta_locations)

us<-map_data('state') %>%
  mutate(
    zcta_region = case_when(
    region %in% c("maryland", "maine", "new hampshire", "vermont",
                  "rhode island", "connecticut", "massachusetts") ~ "New England",
    region %in% c("new york", "pennsylvania", "new jersey", "delaware",
                  "maryland", "district of columbia") ~ "Mid-Atlantic",
    region %in% c("arkansas", "louisiana", "kentucky", "tennessee",
                  "mississippi", "alabama",
                     "georgia", "florida", "south carolina", "north carolina",
                  "west virginia", "virginia") ~ "Southeast",
    region %in% c("montana", "idaho", "wyoming", "nevada", "utah", "colorado") ~ "Rocky Mountains",
    region %in% c("north dakota", "south dakota", "nebraska", "kansas", "minnesota",
                  "iowa","missouri", "wisconsin", "illinois", "michigan", "ohio",
                  "indiana") ~ "Midwest",
    region %in% c("texas", "arizona", "new mexico", "oklahoma") ~ "Southwest",
    region %in% c("washington", "oregon", "california", "alaska", "hawaii") ~ "Pacific Coastal"
    )
  ) %>%
  inner_join(as.data.frame(trends_regionZ))



# zcta = immigrants %>%
#   filter(ZCTAlon > -124.8) %>%
#   select(RID, ZCTA, ZCTAlat, ZCTAlon) %>%
#   mutate(across(c(RID, ZCTA), as.character)) %>%
#   inner_join(mod1_slopes) %>%
#   with_groups(c(ZCTA, ZCTAlon, ZCTAlat),
#               summarise,
#               pefit = mean(estimate),
#               n = n()) %>%
#   ggplot(aes(x = ZCTAlon, y = ZCTAlat)) +
#   geom_polygon(data=us,aes(x=long,y=lat,group=group),
#                color='gray',alpha=.35, inherit.aes = F,
#                fill = NA) +
#   geom_point(aes(color = pefit, size = n), alpha = .7) +
#   scale_x_continuous(breaks = NULL) +
#   scale_y_continuous(breaks = NULL) +
#   scale_color_viridis(option = "plasma", end = 0.8) +
#   scale_fill_viridis(option = "plasma", end = 0.8) +
#   scale_size_continuous(range = c(.5,7))+
#   guides(fill = "none") +
#   labs(x = NULL, y = NULL, n = "Number of immigrants", color = "Average PE-fit") +
```

```r
#   guides(
#     size = "none") +
#   theme_clean()  +
#   theme(legend.position = "bottom")
#
# zcta

zcta = immigrants %>%
  #filter(ZCTAlon > -124.8) %>%
  mutate(across(c(RID, ZCTA), as.character)) %>%
  inner_join(mod1_slopes) %>%
  with_groups(c(ZCTAstate),
              summarise,
              pefit = mean(estimate),
              n = n()) %>%
  filter(n > 50) %>%
  mutate(region = map_chr(ZCTAstate, ~state.name[match(.,state.abb)]),
         region = tolower(region)) %>%
  full_join(map_data("state")) %>%
  ggplot(aes(x=long, y=lat, fill=pefit, group=group)) +
  geom_polygon(color = "white") +
  scale_fill_viridis(option = "plasma", end = 0.8) +
  clean_theme() +
  theme(
    axis.title.x=element_blank(),
    axis.text.x=element_blank(),
    axis.ticks.x=element_blank(),
    axis.title.y=element_blank(),
    axis.text.y=element_blank(),
    axis.ticks.y=element_blank(),
    panel.background = element_rect(fill = "white")) +
  coord_fixed(1.3) +
  labs(title = NULL,
       fill = "PE-fit")

zcta
```

```r
ggarrange(
  ggarrange(ncol = 2, nrow = 3,
            labels = c("A", "B", "C", "D", "E", "F"),
    trend_plot_college + labs(x = "Proportion residents",
                              title = "College degree"),
    trend_plot_foreign2   + labs(x = "Proportion residents",
                                 title = "Foreign born"),
    trend_plot_originContinent + labs(x = "Proportion residents",
                                      title = "From continent of origin"),
    trend_plot_originCountry + labs(x = "Proportion residents",
                                    title = "From country of origin"),
  trend_plot_ancestry + labs(x = "Proportion residents",
                             title = "Same ancestry"),
  trend_plot_regionZ + labs(x = NULL, title = "US Region")),
  zcta,
  nrow = 1,
```

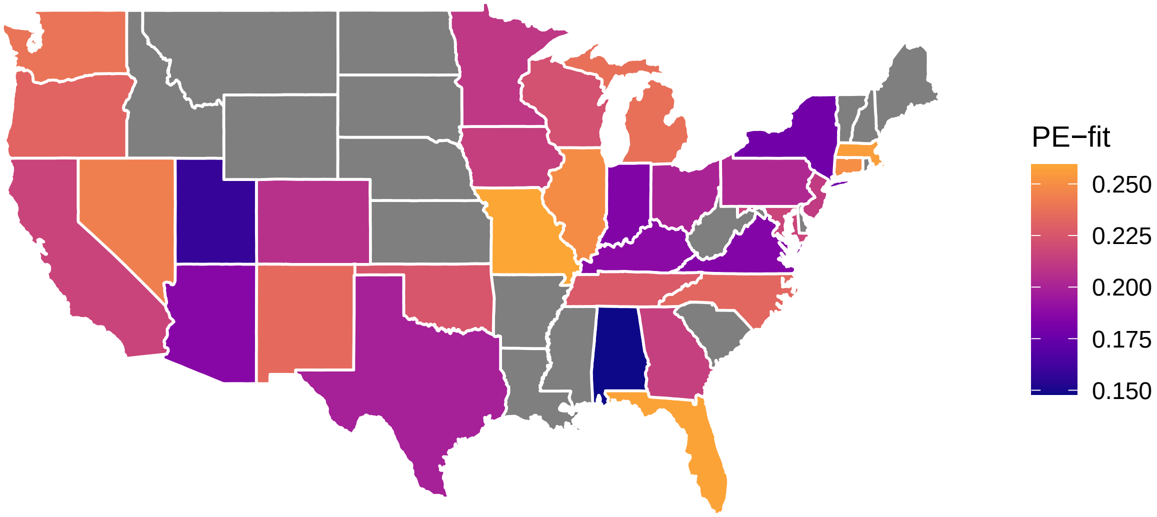
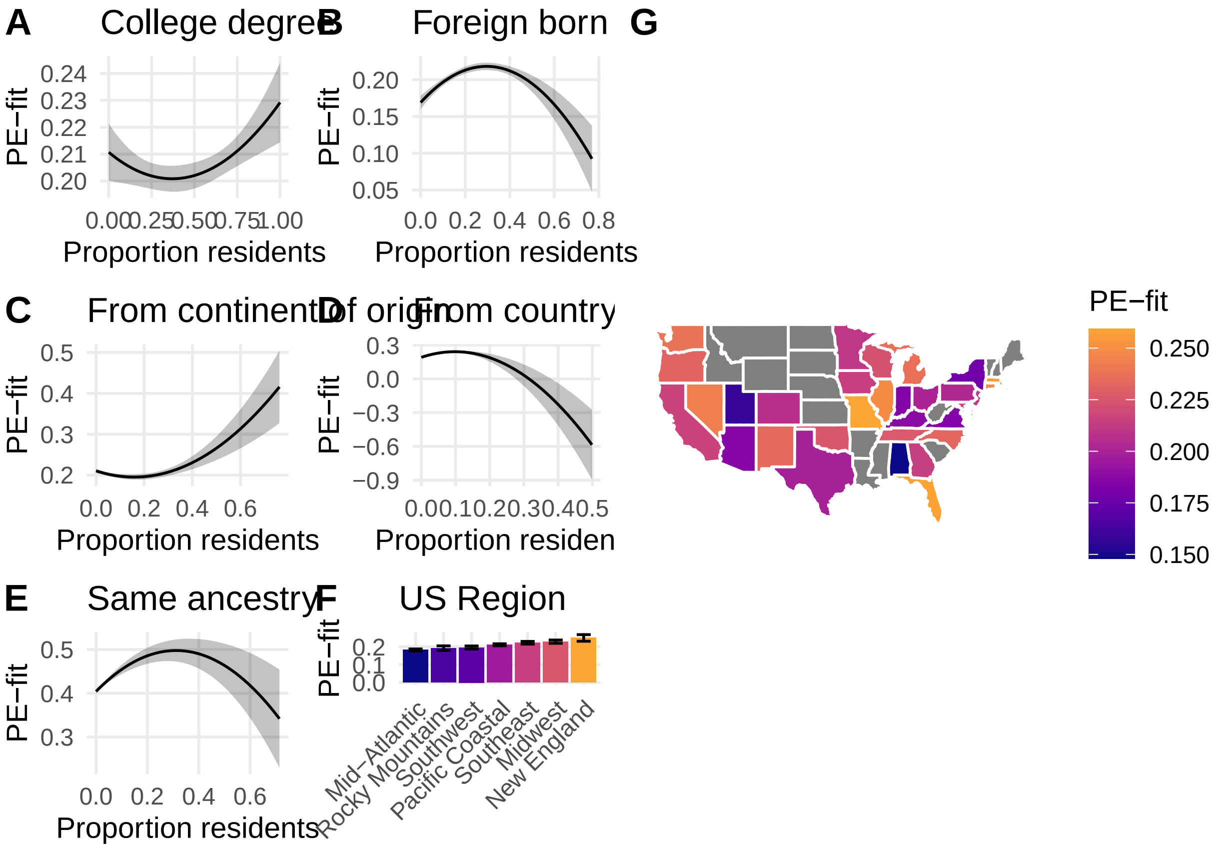


Figure S9: PE-fit across the United States. States are only colored when at least 50 immigrants contribute to the state average.

```
  labels = c(" ", "G")
)
```



```r
ggsave(filename = here("figures/allzcta.jpeg"),
       width = 20, height = 8, units = "in")
```

4.4 RQ4: To what extent does PE-fit improve outcomes for US immigrants?

Finally, we test whether PE-fit among US immigrants is associated with physical and mental health. For these analyses, we anticipate the possibility of a curvilinear relationship, such that not only can too little PE-fit predict worse outcomes, but so can too much PE-fit. Such a relationship cannot be estimated by modifying the multilevel models used to this point. Thus, in order order to allow for these potential curvilinear effects, we extract the estimated slopes for each immigrant from the original model and use these as predictors in subsequent models.

Results can be found in Table S18 and in Figure S10.

After accessing the full data, we will repeat these analyes for self-rated health, stress, and the Ryff well-being scale(s).

```r
#extract PE-fit estimates
immigrants = coef(mod1)$RID %>%
  as.data.frame() %>%
  mutate(RID = rownames(.)) %>%
  select(-`(Intercept)`) %>%
  separate(RID, into = c("RID", "ZCTA")) %>%
  mutate(across(everything(), as.numeric)) %>%
  rename(pe_fit = z_zip_response) %>%
  full_join(immigrants) %>%
  mutate(stress = as.numeric(stress),
         health = as.numeric(health))
```

```r
mod_wb1 = lmer(wellbeing ~ pe_fit + (1 | ZCTA),
               data = immigrants)
mod_wb2 = lmer(wellbeing ~ pe_fit + I(pe_fit^2) + (1 | ZCTA),
               data = immigrants)

mod_stress1 = immigrants %>%
    lmer(stress ~ pe_fit + (1 | ZCTA),
               data = .)
mod_stress2 = immigrants %>%
  lmer(stress ~ pe_fit + I(pe_fit^2) + (1 | ZCTA),
               data = .)

mod_health1 = immigrants %>%
  lmer(health ~ pe_fit + (1 | ZCTA),
               data = .)
mod_health2 = immigrants %>%
  lmer(health ~ pe_fit + I(pe_fit^2) + (1 | ZCTA),
               data = .)


mod_ryff1_1 = lmer(ryff_autonomy ~ pe_fit + (1 | ZCTA),
               data = immigrants)
mod_ryff1_2 = lmer(ryff_autonomy ~ pe_fit + I(pe_fit^2) + (1 | ZCTA),
               data = immigrants)
```

Table S18: Associations of well-being and PE-fit.

	Linear Model	Quadratic Model	
Outcome	PE-fit	PE-fit	PE-fit2
Wellbeing	2.53 (0.06) p < .001	2.69 (0.14) p < .001	-0.41 (0.32) p = .211
Stress	-1.05 (0.08) p < .001	-1.19 (0.20) p < .001	0.35 (0.45) p = .437
Health	1.56 (0.06) p < .001	1.35 (0.15) p < .001	0.51 (0.35) p = .144
Ryff: Autonomy	2.45 (0.48) p < .001	4.36 (1.38) p = .002	-4.42 (2.99) p = .140
Ryff: Environmental Mastery	2.86 (0.51) p < .001	1.52 (1.33) p = .252	3.29 (3.01) p = .274
Ryff: Personal Growth	3.44 (0.45) p < .001	3.41 (1.18) p = .004	0.06 (2.57) p = .981
Ryff: Positive Relations	3.59 (0.48) p < .001	3.28 (1.42) p = .021	0.70 (2.99) p = .815
Ryff: Purpose in Life	3.62 (0.47) p < .001	4.84 (1.15) p < .001	-3.05 (2.62) p = .244
Ryff: Self Acceptance	2.67 (0.47) p < .001	3.10 (1.26) p = .014	-1.01 (2.73) p = .711

```r
mod_ryff2_1 = lmer(ryff_environ_mastery ~ pe_fit + (1 | ZCTA),
                   data = immigrants)
mod_ryff2_2 = lmer(ryff_environ_mastery ~ pe_fit + I(pe_fit^2) + (1 | ZCTA),
                   data = immigrants)

mod_ryff3_1 = lmer(ryff_personal_growth ~ pe_fit + (1 | ZCTA),
                   data = immigrants)
mod_ryff3_2 = lmer(ryff_personal_growth ~ pe_fit + I(pe_fit^2) + (1 | ZCTA),
                   data = immigrants)

mod_ryff4_1 = lmer(ryff_pos_relations ~ pe_fit + (1 | ZCTA),
                   data = immigrants)
mod_ryff4_2 = lmer(ryff_pos_relations ~ pe_fit + I(pe_fit^2) + (1 | ZCTA),
                   data = immigrants)

mod_ryff5_1 = lmer(ryff_purpose ~ pe_fit + (1 | ZCTA),
                   data = immigrants)
mod_ryff5_2 = lmer(ryff_purpose ~ pe_fit + I(pe_fit^2) + (1 | ZCTA),
                   data = immigrants)

mod_ryff6_1 = lmer(ryff_self_accept ~ pe_fit + (1 | ZCTA),
                   data = immigrants)
mod_ryff6_2 = lmer(ryff_self_accept ~ pe_fit + I(pe_fit^2) + (1 | ZCTA),
                   data = immigrants)
```

Test for heteroskedasticity:

```r
mod_wb1_lm = lm(wellbeing ~ pe_fit , data = immigrants)
white_test_boot(mod_wb1_lm)
```

```
## Bootstrapped White's test results
##
## Null hypothesis: Homoskedasticity of the residuals
## Alternative hypothesis: Heteroskedasticity of the residuals
## Number of bootstrap samples: 1000
## Bootstrapped Test Statistic: 73.7
```

Figure S10: Relationship of well-being to PE_fit.

```
## P-value: 0
```

```
mod_wb2_lm = lm(wellbeing ~ pe_fit + I(pe_fit^2) , data = immigrants)
white_test_boot(mod_wb2_lm)
```

```
## Bootstrapped White's test results
##
## Null hypothesis: Homoskedasticity of the residuals
## Alternative hypothesis: Heteroskedasticity of the residuals
## Number of bootstrap samples: 1000
## Bootstrapped Test Statistic: 75.61
## P-value: 0
```

```
mod_stress1_lm = lm(stress ~ pe_fit , data = immigrants)
white_test_boot(mod_stress1_lm)
```

```
## Bootstrapped White's test results
##
## Null hypothesis: Homoskedasticity of the residuals
## Alternative hypothesis: Heteroskedasticity of the residuals
## Number of bootstrap samples: 1000
## Bootstrapped Test Statistic: 34.92
## P-value: 0
```

```
mod_stress2_lm = lm(stress ~ pe_fit + I(pe_fit^2) , data = immigrants)
white_test_boot(mod_stress2_lm)
```

```
## Bootstrapped White's test results
##
## Null hypothesis: Homoskedasticity of the residuals
## Alternative hypothesis: Heteroskedasticity of the residuals
## Number of bootstrap samples: 1000
## Bootstrapped Test Statistic: 34.88
## P-value: 0
```

```
mod_health1_lm = lm(health ~ pe_fit , data = immigrants)
white_test_boot(mod_health1_lm)
```

```
## Bootstrapped White's test results
##
## Null hypothesis: Homoskedasticity of the residuals
## Alternative hypothesis: Heteroskedasticity of the residuals
## Number of bootstrap samples: 1000
## Bootstrapped Test Statistic: 84.27
## P-value: 0
```

```
mod_health2_lm = lm(health ~ pe_fit + I(pe_fit^2) , data = immigrants)
white_test_boot(mod_health2_lm)
```

```
## Bootstrapped White's test results
##
## Null hypothesis: Homoskedasticity of the residuals
## Alternative hypothesis: Heteroskedasticity of the residuals
## Number of bootstrap samples: 1000
## Bootstrapped Test Statistic: 81.85
## P-value: 0
```

```
(wtb_ryff1 = lm(ryff_autonomy ~ pe_fit,
                data = immigrants) %>%
  white_test_boot())
```

```
## Bootstrapped White's test results
##
## Null hypothesis: Homoskedasticity of the residuals
## Alternative hypothesis: Heteroskedasticity of the residuals
## Number of bootstrap samples: 1000
## Bootstrapped Test Statistic: 9.7
## P-value: 0.01
```

```
(wtb_ryff2 = lm(ryff_environ_mastery ~ pe_fit,
                data = immigrants) %>%
  white_test_boot())
```

```
## Bootstrapped White's test results
##
## Null hypothesis: Homoskedasticity of the residuals
## Alternative hypothesis: Heteroskedasticity of the residuals
## Number of bootstrap samples: 1000
## Bootstrapped Test Statistic: 10.24
## P-value: 0.01
```

```
(wtb_ryff3 = lm(ryff_personal_growth ~ pe_fit,
                data = immigrants) %>%
  white_test_boot())
```

```
## Bootstrapped White's test results
##
## Null hypothesis: Homoskedasticity of the residuals
## Alternative hypothesis: Heteroskedasticity of the residuals
## Number of bootstrap samples: 1000
## Bootstrapped Test Statistic: 27.32
## P-value: 0
```

```
(wtb_ryff4 = lm(ryff_pos_relations ~ pe_fit,
                data = immigrants) %>%
  white_test_boot())
```

```
## Bootstrapped White's test results
##
## Null hypothesis: Homoskedasticity of the residuals
```

```
## Alternative hypothesis: Heteroskedasticity of the residuals
## Number of bootstrap samples: 1000
## Bootstrapped Test Statistic: 14.53
## P-value: 0
```

```
(wtb_ryff5 = lm(ryff_purpose ~ pe_fit,
               data = immigrants) %>%
  white_test_boot())
```

```
## Bootstrapped White's test results
##
## Null hypothesis: Homoskedasticity of the residuals
## Alternative hypothesis: Heteroskedasticity of the residuals
## Number of bootstrap samples: 1000
## Bootstrapped Test Statistic: 19.5
## P-value: 0
```

```
(wtb_ryff6 = lm(ryff_self_accept ~ pe_fit,
               data = immigrants) %>%
  white_test_boot())
```

```
## Bootstrapped White's test results
##
## Null hypothesis: Homoskedasticity of the residuals
## Alternative hypothesis: Heteroskedasticity of the residuals
## Number of bootstrap samples: 1000
## Bootstrapped Test Statistic: 19.99
## P-value: 0
```

```
hetero_plot = function(model, title){
  plot = model %>%
    augment %>%
  sample_n(1000) %>%
  ggplot(aes(x = pe_fit, y = .resid)) +
  geom_jitter(alpha = .3,
              height = .1, width = 0) +
  labs(x = "Est PE-fit",
       y = "Residual",
       title = title) +
  theme_clean()

  return(plot)
}

hetero_wb2 = hetero_plot(mod_wb2_lm, "Well-being\n(quadratic)")
hetero_wb1 = hetero_plot(mod_wb1_lm, "Well-being\n(linear)")

hetero_stress2 = hetero_plot(mod_stress2_lm, "Stress\n(quadratic)")
hetero_stress1 = hetero_plot(mod_stress1_lm, "Stress\n(linear)")

hetero_health2 = hetero_plot(mod_health2_lm, "Health\n(quadratic)")
hetero_health1 = hetero_plot(mod_health1_lm, "Health\n(linear)")
```

```
ggarrange(plotlist = list(
  hetero_wb2, hetero_wb1,
  hetero_stress2, hetero_stress1,
  hetero_health2, hetero_health1),
  ncol = 2,
  nrow = 3
          )
```

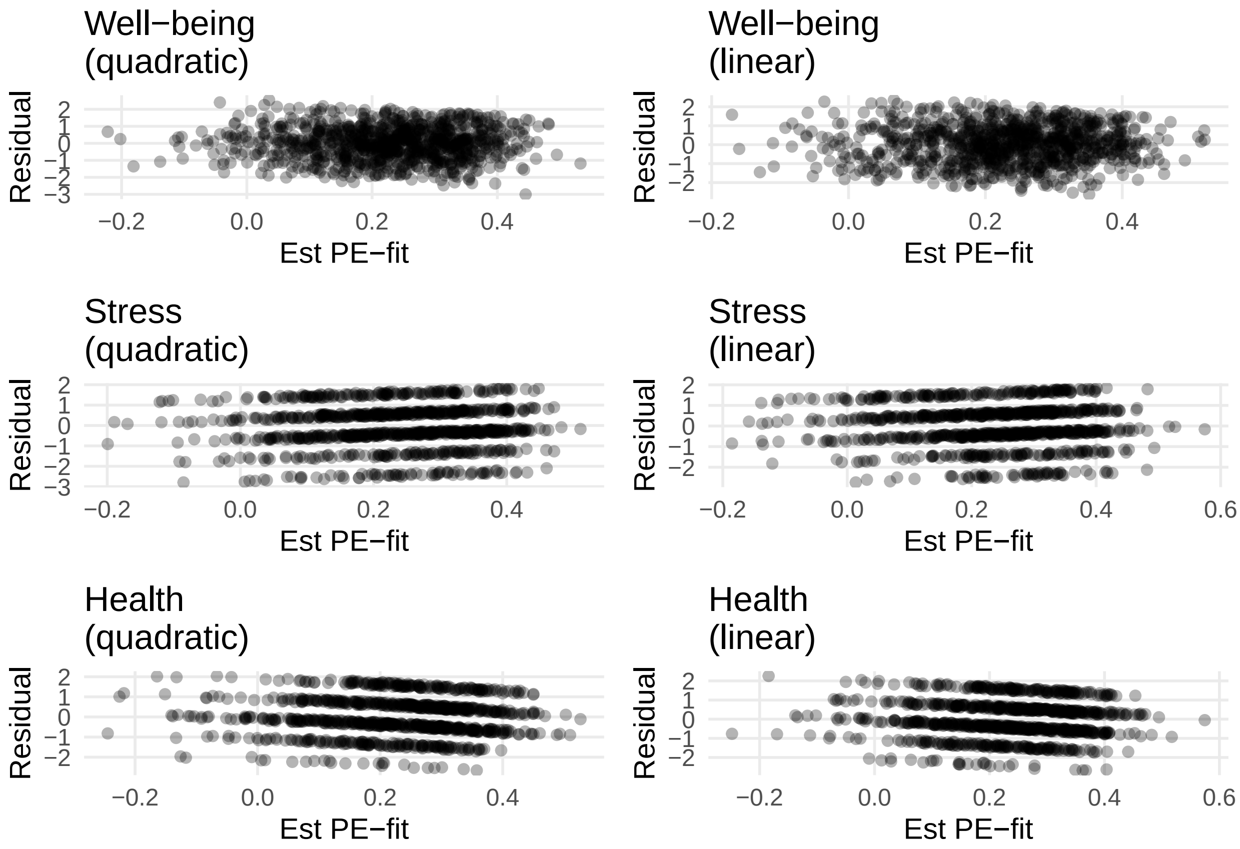


Figure S11: Residuals of well-being-fit models plotted by PE-fit.

4.4.1 Does PE-fit matter more for some?

In our final set of analyses, we investigate whether PE-fit has stronger relationships for well-being for some immigrants – that is, whether the PE-fit/well-being association is moderated by age, education, or region of origin.

```
mod_wb_age = lmer(wellbeing ~ pe_fit*age_group + (1 | ZCTA), data = immigrants)
mod_wb_age_anova = car::Anova(mod_wb_age, type = "III", test.statistic = "F")

mod_stress_age = lmer(stress ~ pe_fit*age_group + (1 | ZCTA), data = immigrants)
mod_stress_age_anova = car::Anova(mod_stress_age, type = "III", test.statistic = "F")

mod_health_age = lmer(health ~ pe_fit*age_group + (1 | ZCTA), data = immigrants)
mod_health_age_anova = car::Anova(mod_health_age, type = "III", test.statistic = "F")
```

```
mod_ryff1_age = lmer(ryff_autonomy ~ pe_fit*age_group + (1 | ZCTA), data = immigrants)

mod_ryff2_age = lmer(ryff_environ_mastery ~ pe_fit*age_group + (1 | ZCTA), data = immigrants)

mod_ryff3_age = lmer(ryff_personal_growth ~ pe_fit*age_group + (1 | ZCTA), data = immigrants)

mod_ryff4_age = lmer(ryff_pos_relations ~ pe_fit*age_group + (1 | ZCTA), data = immigrants)

mod_ryff5_age = lmer(ryff_purpose ~ pe_fit*age_group + (1 | ZCTA), data = immigrants)

mod_ryff6_age = lmer(ryff_self_accept ~ pe_fit*age_group + (1 | ZCTA), data = immigrants)
```

```
models = ls()[str_detect(ls(), "mod_.*_age$")]
tibble(outcome = models) %>%
  mutate(model = purrr::map(outcome, get),
         slopes = purrr::map(model, emtrends,
                             ~age_group,
                             lmerTest.limit = 21013,
                             var = "pe_fit"),
         slopes = purrr::map(slopes, as.data.frame)) %>%
  select(-model) %>%
  unnest(cols = c(slopes)) %>%
  mutate(
    lower.CL = ifelse(is.na(lower.CL), asymp.LCL, lower.CL),
    upper.CL = ifelse(is.na(upper.CL), asymp.UCL, upper.CL),
    sig = case_when(
      lower.CL > 0 ~ "*",
      upper.CL < 0 ~ "*",
      TRUE ~ ""
    ),
    across(where(is.numeric),printnum),
    cell = str_c(pe_fit.trend, sig, "\n[", lower.CL, ", ", upper.CL, "]"),
    outcome = str_remove(outcome, "mod_"),
    outcome = str_remove(outcome, "_age"),
    outcome = factor(outcome,
                     levels = c(
                       "wb", "stress", "health",
                       str_c("ryff", 1:6)),
                     labels = c(
                       "Wellbeing", "Stress", "Health",
                       "Autonomy",
                       "Environmental Mastery",
                       "Personal Growth",
                       "Positive Relations",
                       "Purpose in Life",
                       "Self Acceptance"
                       ))) %>%
  select(outcome, age_group, cell) %>%
  arrange(outcome) %>%
  rename(Age = age_group) %>%
  pivot_wider(names_from = outcome, values_from = cell) %>%
  kable(
    booktabs = T,
```

```
    escape = F,
    caption = "Unstandardized slopes of PE-fit to well-being outcomes, separated by age group") %>%
  kable_classic() %>%
  landscape() %>%
  add_header_above(c(" " = 4, "Ryff Psychological Wellbeing" = 6))
```

Table S19: Unstandardized slopes of PE-fit to well-being outcomes, separated by age group

				Ryff Psychological Wellbeing				
Age	Wellbeing	Stress	Health	Autonomy	Environmental Mastery	Personal Growth	Positive Relations	Purpos
13-17	1.62* [1.37, 1.86]	-0.71* [-1.07, -0.35]	1.23* [0.96, 1.50]	1.26 [-1.21, 3.74]	1.63 [-1.03, 4.28]	2.39* [0.16, 4.62]	4.32* [1.77, 6.86]	4.46* [
18-25	2.14* [1.98, 2.31]	-0.89* [-1.13, -0.65]	1.40* [1.21, 1.58]	2.47* [0.98, 3.97]	1.78* [0.21, 3.35]	3.99* [2.54, 5.44]	3.06* [1.57, 4.54]	3.54* [
26-35	2.36* [2.10, 2.62]	-1.23* [-1.61, -0.84]	1.42* [1.14, 1.71]	2.00 [-0.26, 4.26]	1.42 [-0.80, 3.65]	1.45 [-0.66, 3.55]	2.41* [0.29, 4.53]	2.36* [
36-45	2.29* [1.89, 2.69]	-1.06* [-1.61, -0.50]	1.52* [1.08, 1.96]	-1.02 [-4.58, 2.54]	0.11 [-3.41, 3.64]	4.98* [1.69, 8.27]	5.19* [1.52, 8.85]	2.89* [
46-55	2.27* [1.74, 2.80]	-0.38 [-1.10, 0.34]	1.08* [0.51, 1.66]	3.54 [-0.02, 7.09]	6.55* [2.35, 10.75]	0.55 [-3.03, 4.13]	4.85* [0.36, 9.34]	2.18 [-
56-65	2.05* [1.28, 2.82]	-0.54 [-1.57, 0.48]	1.94* [1.10, 2.78]	4.33 [-3.58, 12.23]	5.62* [1.11, 10.14]	4.04 [-1.25, 9.33]	3.04 [-2.32, 8.41]	5.24 [-
66+	4.28* [3.03, 5.53]	-0.90 [-2.72, 0.91]	2.20* [0.84, 3.56]	-3.21 [-12.93, 6.52]	3.08 [-9.00, 15.16]	4.25 [-3.28, 11.79]	-3.46 [-12.29, 5.38]	-1.50 [

```
mod_wb_edu = lmer(wellbeing ~ pe_fit*education + (1 | ZCTA), data = immigrants)
mod_wb_edu_anova = car::Anova(mod_wb_edu, type = "III", test.statistic = "F")

mod_stress_edu = lmer(stress ~ pe_fit*education + (1 | ZCTA), data = immigrants)
mod_stress_edu_anova = car::Anova(mod_stress_edu, type = "III", test.statistic = "F")

mod_health_edu = lmer(health ~ pe_fit*education + (1 | ZCTA), data = immigrants)
mod_health_edu_anova = car::Anova(mod_health_edu, type = "III", test.statistic = "F")

mod_ryff1_edu = lmer(ryff_autonomy ~ pe_fit*education + (1 | ZCTA), data = immigrants)

mod_ryff2_edu = lmer(ryff_environ_mastery ~ pe_fit*education + (1 | ZCTA), data = immigrants)

mod_ryff3_edu = lmer(ryff_personal_growth ~ pe_fit*education + (1 | ZCTA), data = immigrants)

mod_ryff4_edu = lmer(ryff_pos_relations ~ pe_fit*education + (1 | ZCTA), data = immigrants)

mod_ryff5_edu = lmer(ryff_purpose ~ pe_fit*education + (1 | ZCTA), data = immigrants)

mod_ryff6_edu = lmer(ryff_self_accept ~ pe_fit*education + (1 | ZCTA), data = immigrants)
```

```
models = ls()[str_detect(ls(), "mod_.*_edu$")]
tibble(outcome = models) %>%
  mutate(model = purrr::map(outcome, get),
         slopes = purrr::map(model, emtrends,
                             ~education,
                             lmerTest.limit = 19316,
                             var = "pe_fit"),
         slopes = purrr::map(slopes, as.data.frame)) %>%
  select(-model) %>%
  unnest(cols = c(slopes)) %>%
  mutate(
    lower.CL = ifelse(is.na(lower.CL), asymp.LCL, lower.CL),
    upper.CL = ifelse(is.na(upper.CL), asymp.UCL, upper.CL),
    sig = case_when(
      lower.CL > 0 ~ "*",
      upper.CL < 0 ~ "*",
      TRUE ~ ""
    ),
    across(where(is.numeric),printnum),
    cell = str_c(pe_fit.trend, sig, " [", lower.CL, ", ", upper.CL, "]"),
    outcome = str_remove(outcome, "mod_"),
    outcome = str_remove(outcome, "_edu"),
    outcome = factor(outcome,
                 levels = c(
                   "wb", "stress", "health",
                   str_c("ryff", 1:6)),
                 labels = c(
                   "Wellbeing", "Stress", "Health",
                   "Autonomy",
                   "Environmental Mastery",
                   "Personal Growth",
                   "Positive Relations",
```

```
                    "Purpose in Life",
                    "Self Acceptance"
                    ))) %>%
  select(outcome, education, cell) %>%
  arrange(outcome) %>%
  pivot_wider(names_from = outcome, values_from = cell) %>%
  kable(
    booktabs = T,
    escape = F,
    caption = "Unstandardized slopes of PE-fit to well-being outcomes, separated by educational attainme
  kable_classic() %>%
  landscape() %>%
  add_header_above(c(" " = 4, "Ryff Psychological Well Being" = 6))
```

Table S20: Unstandardized slopes of PE-fit to well-being outcomes, separated by educational attainment

				Ryff Psychological Well Being			
education	Wellbeing	Stress	Health	Autonomy	Environmental Mastery	Personal Growth	Positive Rela
less12yrs	1.39* [1.03, 1.75]	-0.32 [-0.84, 0.19]	0.93* [0.55, 1.32]	1.97 [-1.25, 5.19]	3.02 [-0.89, 6.94]	4.41* [1.48, 7.34]	1.40 [-2.49, 5.
HSgrad	2.03* [1.74, 2.32]	-0.88* [-1.31, -0.46]	1.35* [1.04, 1.67]	0.51 [-2.82, 3.83]	0.45 [-2.45, 3.35]	3.17* [0.90, 5.45]	4.28* [1.10, 7
CurrentInUniv	2.36* [2.14, 2.58]	-1.03* [-1.35, -0.72]	1.28* [1.04, 1.52]	1.70 [-0.19, 3.59]	2.28* [0.27, 4.28]	2.42* [0.53, 4.31]	4.11* [2.25, 5
SomeCollege	3.17* [2.63, 3.72]	-1.37* [-2.13, -0.60]	1.58* [0.99, 2.17]	4.32 [-0.53, 9.16]	1.16 [-3.73, 6.06]	6.89* [3.25, 10.53]	1.84 [-2.40, 6.
AssociateDegree	2.45* [1.68, 3.22]	-1.77* [-2.96, -0.59]	2.06* [1.23, 2.89]	4.89 [-0.36, 10.13]	4.26 [-1.65, 10.17]	-0.55 [-9.02, 7.91]	1.77 [-5.30, 8.
CollegeDegree	2.39* [2.11, 2.67]	-0.85* [-1.25, -0.45]	1.24* [0.94, 1.54]	2.13 [-0.44, 4.71]	2.69 [-0.11, 5.49]	3.07* [0.47, 5.67]	4.65* [1.99, 7
InGradOrProSchool	2.80* [2.27, 3.33]	-0.79* [-1.54, -0.05]	1.35* [0.78, 1.92]	-2.20 [-7.91, 3.50]	5.21 [-0.14, 10.57]	0.80 [-4.41, 6.02]	1.02 [-4.54, 6.
GradOrProDegree	2.50* [2.23, 2.78]	-1.32* [-1.71, -0.93]	1.84* [1.54, 2.14]	3.39* [1.22, 5.56]	3.41* [0.94, 5.87]	2.67* [0.51, 4.82]	2.71* [0.63, 4

```r
mod_wb_region = lmer(wellbeing ~ pe_fit*region + (1 | ZCTA), data = immigrants)
mod_wb_region_anova = car::Anova(mod_wb_region, type = "III", test.statistic = "F")

mod_stress_region = lmer(stress ~ pe_fit*region + (1 | ZCTA), data = immigrants)
mod_stress_region_anova = car::Anova(mod_stress_region, type = "III", test.statistic = "F")

mod_health_region = lmer(health ~ pe_fit*region + (1 | ZCTA), data = immigrants)
mod_health_region_anova = car::Anova(mod_health_region, type = "III", test.statistic = "F")

mod_ryff1_region = lmer(ryff_autonomy ~ pe_fit*region + (1 | ZCTA), data = immigrants)

mod_ryff2_region = lmer(ryff_environ_mastery ~ pe_fit*region + (1 | ZCTA), data = immigrants)

mod_ryff3_region = lmer(ryff_personal_growth ~ pe_fit*region + (1 | ZCTA), data = immigrants)

mod_ryff4_region = lmer(ryff_pos_relations ~ pe_fit*region + (1 | ZCTA), data = immigrants)

mod_ryff5_region = lmer(ryff_purpose ~ pe_fit*region + (1 | ZCTA), data = immigrants)

mod_ryff6_region = lmer(ryff_self_accept ~ pe_fit*region + (1 | ZCTA), data = immigrants)
```

```r
models = ls()[str_detect(ls(), "mod_.*_region$")]
tibble(outcome = models) %>%
  mutate(model = purrr::map(outcome, get),
         slopes = purrr::map(model, emtrends,
                             ~region,
                             lmerTest.limit = 21009,
                             var = "pe_fit"),
         slopes = purrr::map(slopes, as.data.frame)) %>%
  select(-model) %>%
  unnest(cols = c(slopes)) %>%
  mutate(
    lower.CL = ifelse(is.na(lower.CL), asymp.LCL, lower.CL),
    upper.CL = ifelse(is.na(upper.CL), asymp.UCL, upper.CL),
    sig = case_when(
      lower.CL > 0 ~ "*",
      upper.CL < 0 ~ "*",
      TRUE ~ ""
    ),
    across(where(is.numeric),printnum),
    cell = str_c(pe_fit.trend, sig, " [", lower.CL, ", ", upper.CL, "]"),
    outcome = str_remove(outcome, "mod_"),
    outcome = str_remove(outcome, "_region"),
    outcome = factor(outcome,
                     levels = c(
                       "wb", "stress", "health",
                       str_c("ryff", 1:6)),
                     labels = c(
                       "Wellbeing", "Stress", "Health",
                       "Autonomy",
                       "Environmental Mastery",
                       "Personal Growth",
                       "Positive Relations",
```

```
                    "Purpose in Life",
                    "Self Acceptance"
                    ))) %>%
  select(outcome, region, cell) %>%
  arrange(outcome) %>%
  pivot_wider(names_from = outcome, values_from = cell) %>%
  kable(
    booktabs = T,
    escape = F,
    caption = "Unstandardized slopes of PE-fit to well-being outcomes, separated by region of origin")
  kable_classic() %>%
  landscape() %>%
  add_header_above(c(" " = 4, "Ryff Psychological Well Being" = 6))
```

Table S21: Unstandardized slopes of PE-fit to well-being outcomes, separated by region of origin

				Ryff Psychlogical Well B			
region	Wellbeing	Stress	Health	Autonomy	Environmental Mastery	Personal Growth	Positi
Africa	2.48* [2.20, 2.75]	-1.25* [-1.66, -0.85]	1.15* [0.85, 1.44]	2.42 [-0.62, 5.46]	0.69 [-2.60, 3.99]	4.82* [2.21, 7.44]	4.43*
East Asia	2.30* [1.76, 2.84]	-1.08* [-1.86, -0.29]	1.04* [0.47, 1.61]	1.66 [-2.56, 5.88]	3.24 [-0.56, 7.04]	2.16 [-0.98, 5.30]	5.03*
Europe	2.38* [2.07, 2.70]	-0.34 [-0.77, 0.09]	1.22* [0.88, 1.55]	3.56* [1.10, 6.01]	5.79* [2.59, 8.99]	1.22 [-1.08, 3.52]	2.40 [
Middle East	2.29* [1.81, 2.76]	-0.46 [-1.13, 0.21]	1.20* [0.69, 1.71]	2.30 [-1.01, 5.61]	1.93 [-1.88, 5.74]	5.25* [1.53, 8.97]	4.61*
North America	2.03* [1.43, 2.63]	-0.06 [-0.86, 0.73]	1.01* [0.37, 1.64]	-2.24 [-6.58, 2.10]	4.00 [-0.14, 8.13]	2.07 [-0.98, 5.11]	3.48 [
Oceania	2.29* [1.49, 3.10]	-0.94 [-2.05, 0.18]	1.19* [0.32, 2.07]	-5.87 [-11.92, 0.19]	4.84 [-0.63, 10.32]	-0.47 [-4.68, 3.74]	3.38 [
South and Central America	3.59* [3.31, 3.86]	-1.47* [-1.86, -1.09]	1.75* [1.46, 2.05]	2.52* [0.27, 4.76]	4.38* [1.95, 6.82]	4.39* [1.98, 6.80]	2.79*
South/Southeast Asia	2.16* [1.98, 2.33]	-1.02* [-1.28, -0.76]	1.34* [1.16, 1.53]	2.49* [0.89, 4.09]	0.93 [-0.70, 2.56]	3.87* [2.20, 5.53]	3.43*

Table S22: Is the relationship of PE-fit to well-being moderated by immigrant characteristics? Statistical tests of moderation terms.

outcome	age	edu	region
health	F(6,19,857.08) = 2.16, p = .044	F(7,19,250.21) = 3.45, p = .001	F(7,20,582.81) = 2.08, p = .043
ryff1	F(6,778.23) = 0.90, p = .496	F(7,758.20) = 1.05, p = .391	F(7,804.20) = 1.85, p = .075
ryff2	F(6,745.60) = 2.04, p = .058	F(7,725.74) = 0.65, p = .716	F(7,767.59) = 2.75, p = .008
ryff3	F(6,770.29) = 2.28, p = .035	F(7,733.76) = 1.93, p = .063	F(7,843.49) = 1.58, p = .137
ryff4	F(6,703.21) = 1.52, p = .170	F(7,808.48) = 2.09, p = .043	F(7,865.70) = 0.84, p = .557
ryff5	F(6,699.63) = 0.91, p = .485	F(7,717.91) = 1.96, p = .058	F(7,749.09) = 1.55, p = .147
ryff6	F(6,841.36) = 2.51, p = .021	F(7,826.90) = 0.77, p = .610	F(7,858.61) = 1.29, p = .251
stress	F(6,12,472.56) = 2.23, p = .037	F(7,12,578.84) = 2.48, p = .015	F(7,13,283.50) = 3.75, p < .001
wb	F(6,20,063.18) = 6.20, p < .001	F(7,19,527.46) = 6.16, p < .001	F(7,20,933.26) = 12.53, p < .001

```r
wellbeing_mods = data.frame(
  name = c(mod_age, mod_edu, mod_region)
) %>%
  mutate(
    model = purrr::map(name, get),
    aov = purrr::map(model, car::Anova,
                     type = "III",
                     test.statistic = "F"))
```

```r
wellbeing_mods %>%
  select(name, aov) %>%
  mutate(aov = purrr::map(aov, tidy)) %>%
  unnest(c(aov)) %>%
  filter(str_detect(term, ":")) %>%
  mutate(p.value = printp(p.value),
         p.value = ifelse(str_detect(p.value, "<"),
                          p.value,
                          str_c("= ", p.value)),
         across(c(statistic, Df.res), printnum),
         cell = str_c("F(", df, ",", Df.res,") = ", statistic, ", p ", p.value)) %>%
  select(name, cell) %>%
  mutate(name = str_remove(name, "mod_")) %>%
  separate(name, into = c("outcome", "moderator")) %>%
  pivot_wider(names_from = moderator, values_from = cell) %>%
  kable(
    booktabs = T,
    escape = F,
    caption = "Is the relationship of PE-fit to well-being moderated by immigrant characteristics? Stati
  kable_classic()
```

```r
models = ls()[str_detect(ls(), "mod_.*_age$")]
models = models[!str_detect(models, "ryff")]

models_age = tibble(outcome = models) %>%
  mutate(model = purrr::map(outcome, get),
         slopes = purrr::map(model, emtrends,
```

```
                             ~age_group,
                             lmerTest.limit = 21013,
                             var = "pe_fit"),
         slopes = purrr::map(slopes, as.data.frame)) %>%
  select(-model) %>%
  unnest(cols = c(slopes)) %>%
  rename(group = age_group) %>%
  mutate(moderator = "Age")

models = ls()[str_detect(ls(), "mod_.*_edu$")]
models = models[!str_detect(models, "ryff")]

models_edu = tibble(outcome = models) %>%
  mutate(model = purrr::map(outcome, get),
         slopes = purrr::map(model, emtrends,
                             ~education,
                             lmerTest.limit = 21013,
                             var = "pe_fit"),
         slopes = purrr::map(slopes, as.data.frame)) %>%
  select(-model) %>%
  unnest(cols = c(slopes)) %>%
  rename(group = education) %>%
  mutate(moderator = "Education")

models = ls()[str_detect(ls(), "mod_.*_region$")]
models = models[!str_detect(models, "ryff")]

models_region = tibble(outcome = models) %>%
  mutate(model = purrr::map(outcome, get),
         slopes = purrr::map(model, emtrends,
                             ~region,
                             lmerTest.limit = 21013,
                             var = "pe_fit"),
         slopes = purrr::map(slopes, as.data.frame)) %>%
  select(-model) %>%
  unnest(cols = c(slopes)) %>%
  rename(group = region) %>%
  mutate(moderator = "Region")

models_age %>%
  full_join(models_edu) %>%
  full_join(models_region) %>%
  mutate(
    lower.CL = ifelse(is.na(lower.CL), asymp.LCL, lower.CL),
    upper.CL = ifelse(is.na(upper.CL), asymp.UCL, upper.CL),
    outcome = str_remove(outcome, "mod_"),
    outcome = str_remove(outcome, "_age"),
    outcome = str_remove(outcome, "_edu"),
    outcome = str_remove(outcome, "_region"),
    outcome = factor(outcome,
                     levels = c(
                       "wb", "stress", "health",
                       str_c("ryff", 1:6)),
                     labels = c(
```

```
                        "Wellbeing", "Stress", "Health",
                        "Autonomy",
                        "Environmental Mastery",
                        "Personal Growth",
                        "Positive Relations",
                        "Purpose in Life",
                        "Self Acceptance"
                      ))) %>%
  ggplot(aes(x = group, y = pe_fit.trend)) +
  # geom_point(aes(color = outcome)) +# geom_line(aes(color = outcome)) +
  geom_bar(stat = "identity", aes(fill = outcome),
           color = "white",
           alpha = .7) +
  geom_errorbar(aes(ymin = lower.CL, ymax = upper.CL),
                color = "black",
                width = .5) +
  geom_hline(aes(yintercept = 0)) +
  guides(color = "none", fill = "none") +
  scale_fill_viridis_d(option = "plasma", end = 0.8) +
  #scale_x_discrete(labels = label_wrap(10)) +
  facet_grid(outcome~moderator, scales = "free") +
  labs(x = NULL, y = "Estimated relationship between PE-fit and outcome") +
  theme_clean() +
  theme(axis.text.x = element_text(angle = 30, hjust = 1, vjust = 1))
```

```
ggsave(filename = here("figures/wb_mods.pdf"), width = 8*1.5, height = 4*1.5)
```

4.4.2 Controlling for big five traits

Because personality traits are associated with PE-fit, it may be that associations between fit and well-being and driven by traits as a third variable. To test for this, we run additional models controlling for traits to see if the effect of PE-fit persists.

Note that these analyses will only be conducted on the Ryff PWB measure, not the SPI measure of wellbeing, as the later has item overlap with the measure of neuroticism.

```
immigrants = immigrants %>%
  mutate(
    across(
      starts_with("SPI_"),
      z_fun
    ))

mod_wb_person_1 = lmer(wellbeing ~ pe_fit +
                         SPI_Agree + SPI_Consc + SPI_Extra + SPI_Neuro + SPI_Open +
                         (1 | ZCTA),
                     data = immigrants)

mod_wb_person_2 = lmer(wellbeing ~ pe_fit + I(pe_fit^2) +
                         SPI_Agree + SPI_Consc + SPI_Extra + SPI_Neuro + SPI_Open +
                         (1 | ZCTA),
                     data = immigrants)
```

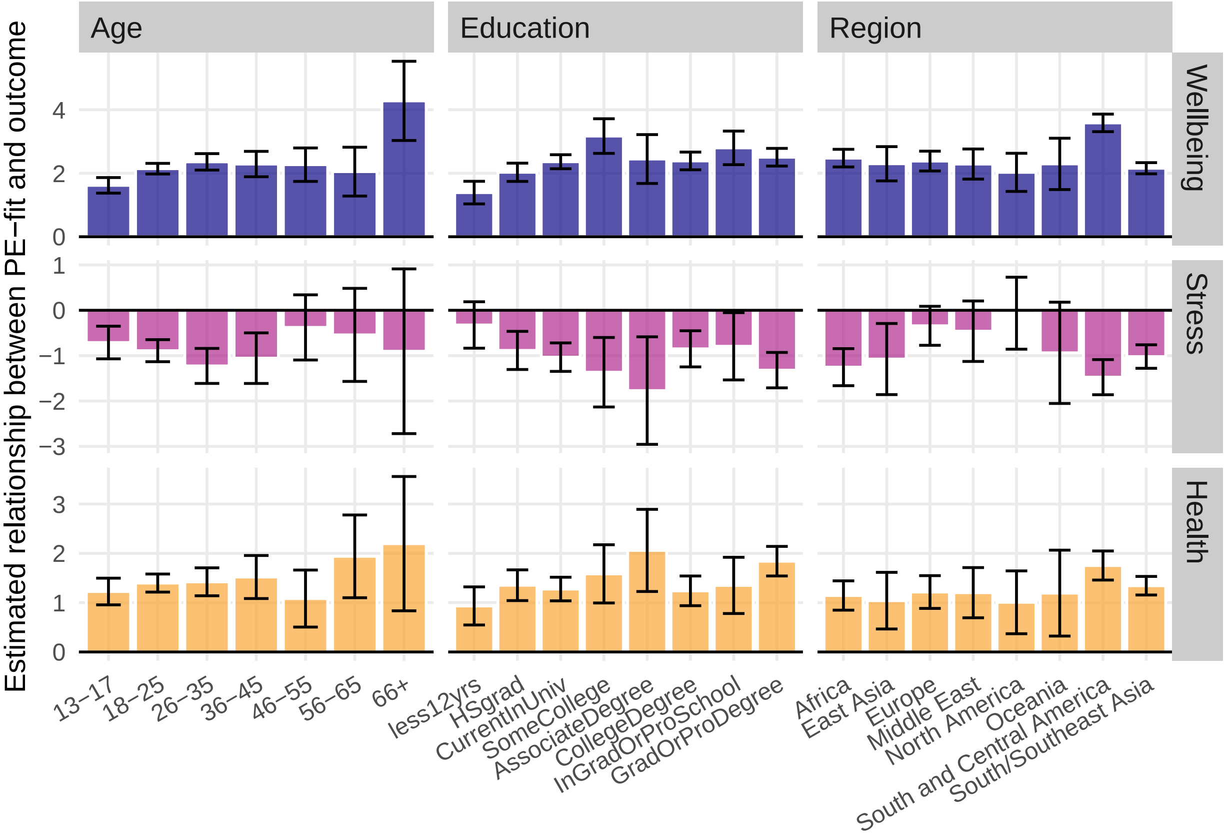


Figure S12: Estimated simple slopes of well-being to PE-fit relationships.

```
mod_stress_person_1 = lmer(stress ~ pe_fit +
                     SPI_Agree + SPI_Consc + SPI_Extra + SPI_Neuro + SPI_Open +
                     (1 | ZCTA),
               data = immigrants)

mod_stress_person_2 = lmer(stress ~ pe_fit + I(pe_fit^2) +
                     SPI_Agree + SPI_Consc + SPI_Extra + SPI_Neuro + SPI_Open +
                     (1 | ZCTA),
               data = immigrants)

mod_health_person_1 = lmer(health ~ pe_fit +
                     SPI_Agree + SPI_Consc + SPI_Extra + SPI_Neuro + SPI_Open +
                     (1 | ZCTA),
               data = immigrants)

mod_health_person_2 = lmer(health ~ pe_fit + I(pe_fit^2) +
                     SPI_Agree + SPI_Consc + SPI_Extra + SPI_Neuro + SPI_Open +
                     (1 | ZCTA),
               data = immigrants)

mod_ryff1_person_1 = lmer(ryff_autonomy ~ pe_fit +
                     SPI_Agree + SPI_Consc + SPI_Extra + SPI_Neuro + SPI_Open +
                     (1 | ZCTA),
               data = immigrants)
mod_ryff1_person_2 = lmer(ryff_autonomy ~ pe_fit + I(pe_fit^2) +
                     SPI_Agree + SPI_Consc + SPI_Extra + SPI_Neuro + SPI_Open +
                     (1 | ZCTA),
               data = immigrants)

mod_ryff2_person_1 = lmer(ryff_environ_mastery ~ pe_fit +
                     SPI_Agree + SPI_Consc + SPI_Extra + SPI_Neuro + SPI_Open +
                     (1 | ZCTA),
               data = immigrants)
mod_ryff2_person_2 = lmer(ryff_environ_mastery ~ pe_fit + I(pe_fit^2) +
                     SPI_Agree + SPI_Consc + SPI_Extra + SPI_Neuro + SPI_Open +
                     (1 | ZCTA),
               data = immigrants)

mod_ryff3_person_1 = lmer(ryff_personal_growth ~ pe_fit +
                     SPI_Agree + SPI_Consc + SPI_Extra + SPI_Neuro + SPI_Open +
                     (1 | ZCTA),
               data = immigrants)
mod_ryff3_person_2 = lmer(ryff_personal_growth ~ pe_fit + I(pe_fit^2) +
                     SPI_Agree + SPI_Consc + SPI_Extra + SPI_Neuro + SPI_Open +
                     (1 | ZCTA),
               data = immigrants)

mod_ryff4_person_1 = lmer(ryff_pos_relations ~ pe_fit +
                     SPI_Agree + SPI_Consc + SPI_Extra + SPI_Neuro + SPI_Open +
                     (1 | ZCTA),
               data = immigrants)
mod_ryff4_person_2 = lmer(ryff_pos_relations ~ pe_fit + I(pe_fit^2) +
                     SPI_Agree + SPI_Consc + SPI_Extra + SPI_Neuro + SPI_Open +
```

```
                     (1 | ZCTA),
               data = immigrants)

mod_ryff5_person_1 = lmer(ryff_purpose ~ pe_fit +
                    SPI_Agree + SPI_Consc + SPI_Extra + SPI_Neuro + SPI_Open +
                            (1 | ZCTA),
               data = immigrants)
mod_ryff5_person_2 = lmer(ryff_purpose ~ pe_fit + I(pe_fit^2) + SPI_Agree + SPI_Consc + SPI_Extra + SPI_
                     (1 | ZCTA),
               data = immigrants)

mod_ryff6_person_1 = lmer(ryff_self_accept ~ pe_fit +
                     SPI_Agree + SPI_Consc + SPI_Extra + SPI_Neuro + SPI_Open +
                     (1 | ZCTA),
               data = immigrants)
mod_ryff6_person_2 = lmer(ryff_self_accept ~ pe_fit + I(pe_fit^2) +
                     SPI_Agree + SPI_Consc + SPI_Extra + SPI_Neuro + SPI_Open +
                     (1 | ZCTA),
               data = immigrants)
```

Table S23: Relationship of PE-fit to well-being after controlling for Big Five

				Ryff				
term	WellBeing	Stress	Health	Autonomy	Environmental Mastery	Personal Growth	Positive Relations	Purpos
PE-fit	0.78*** (0.05)	-0.42*** (0.09)	0.56*** (0.07)	1.91** (0.59)	0.28 (0.60)	0.76 (0.58)	1.09 (0.57)	2.07**
Agreeableness	0.08*** (0.01)	0.02* (0.01)	0.02** (0.01)	-0.26*** (0.06)	0.09 (0.07)	0.11 (0.06)	0.39*** (0.07)	-0.03 (
Conscientiousness	0.18*** (0.01)	-0.06*** (0.01)	0.14*** (0.01)	0.03 (0.06)	0.39*** (0.06)	0.18** (0.06)	0.09 (0.06)	0.28**
Extraversion	0.17*** (0.01)	-0.05*** (0.01)	0.13*** (0.01)	0.05 (0.06)	0.21*** (0.06)	0.09 (0.05)	0.35*** (0.06)	0.08 (
Neuroticism	-0.51*** (0.01)	0.46*** (0.01)	-0.26*** (0.01)	-0.30*** (0.06)	-0.51*** (0.06)	-0.28*** (0.05)	-0.24*** (0.06)	-0.18**
Openness	0.04*** (0.01)	0.03** (0.01)	0.01 (0.01)	0.35*** (0.06)	0.11 (0.06)	0.36*** (0.06)	-0.06 (0.06)	0.10 (

Table S24: Relationship of PE-fit to well-being after controlling for Big Five

			Ryff					
term	WellBeing	Stress	Health	Autonomy	Environmental Mastery	Personal Growth	Positive Relations	Purpo
PE-fit	1.02*** (0.11)	-0.83*** (0.19)	0.35* (0.15)	3.56* (1.40)	-1.39 (1.30)	0.78 (1.20)	0.37 (1.38)	2.68* (
PE-fit2	-0.61* (0.24)	1.00* (0.41)	0.51 (0.33)	-3.70 (2.85)	3.95 (2.73)	-0.05 (2.43)	1.59 (2.76)	-1.44 (
Agreeableness	0.08*** (0.01)	0.02* (0.01)	0.02** (0.01)	-0.26*** (0.06)	0.09 (0.07)	0.11 (0.07)	0.39*** (0.07)	-0.03 (
Conscientiousness	0.18*** (0.01)	-0.05*** (0.01)	0.14*** (0.01)	0.02 (0.06)	0.39*** (0.06)	0.18** (0.06)	0.09 (0.06)	0.28**
Extraversion	0.17*** (0.01)	-0.05*** (0.01)	0.13*** (0.01)	0.05 (0.06)	0.21*** (0.06)	0.09 (0.05)	0.35*** (0.06)	0.07 (
Neuroticism	-0.51*** (0.01)	0.46*** (0.01)	-0.26*** (0.01)	-0.30*** (0.05)	-0.50*** (0.06)	-0.28*** (0.05)	-0.24*** (0.06)	-0.18**
Openness	0.04*** (0.01)	0.03** (0.01)	0.02* (0.01)	0.35*** (0.06)	0.12 (0.06)	0.36*** (0.06)	-0.06 (0.06)	0.10 (

5 Sensitivity analysis

5.1 Profiles based on narrow traits

A key concern in developing the current study was whether to create person and ZCTA profiles using items or trait scores. Each method has a key limitation:

- Profiles based on **items** will reflect, in part, the structure of the measurement framework. The SPI measure in particular was derived with the goal of having unidimensional factors – this, we expect items to cluster together.
- Profiles based on **factors** would be grounded in assumptions of measurement equivalence. That is, the items for a particular trait (e.g., Perfectionism) would be assumed to relate identically to the latent factor for all individuals. Given prior work suggesting non-equivalence of the Big-Five across cultures, this assumption seems unlikley to be true. Notably, the equivalence SPI27 across culture or geography has not been tested, although the measurement framework was derived from an international sample.

Both methods are limited, but it is unclear to the authors whether or how these limitations may impact profile similarity. In this section, we estimate profiles using the method not chosen for this project. Then we (1) compare profiles derived from both methods to determine how similar the results are and (2) re-estimate the intial PE-fit model (i.e., what is the PE-fit of US immigrants) to determine how much this model changes.

First, we must estimate profiles for each ZCTA based on the narrow SPI factors. We use all factors except for well-being, as it is an outcome in subsequent analyses.

```r
zip_profiles = us_data %>%
  group_by(ZCTA) %>%
  summarise(
    across( all_of(spi_names), mean, na.rm=T), # mean of each personality item
    n_zip = n() # number of participants
  ) %>%
  ungroup() %>%
  filter(n_zip >= 30)


zip_profile_long = zip_profiles %>%
  #transform to long form
  pivot_longer(
    # name of column storing variable names
    names_to = "trait",
    # name of column storing values
    values_to = "zip_response",
    # which variables to stack
    cols = all_of(spi_names))
```

Next we create a long-form dataset for immigrants, similar to the one used with items, but this time with traits.

```r
immigrants_profile_trait = immigrants %>%
  select(-starts_with("q_")) %>%
  #transform to long form
  pivot_longer(
    # name of column storing variable names
    names_to = "trait",
```

```
    # name of column storing values
    values_to = "response",
    # which variables to stack
    cols = all_of(spi_names)) %>%
  filter(!is.na(response)) %>%
  #merge with location profiles
  left_join(zip_profile_long) %>%
  filter(!is.na(zip_response)) %>%
  # standardize responses within profile
  with_groups(RID, mutate, across(
    # mutate these columns
    contains("response"),
    # with the z-score function
    z_fun,
    # create new name
    .names = "z_{.col}")) %>%
  mutate(status = "Immigrant",
         RID = as.character(RID))
```

Finally, we reestimate the normative profile.

```
us_profile_trait = us_data %>%
  select(-starts_with("q_")) %>%
  #transform to long form
  pivot_longer(
    # name of column storing variable names
    names_to = "trait",
    # name of column storing values
    values_to = "response",
    # which variables to stack
    cols = all_of(spi_names)) %>%
  filter(!is.na(response)) %>%
  # standardize responses within profile
  with_groups(RID, mutate, across(
    # mutate these columns
    contains("response"),
    # with the z-score function
    z_fun,
    # create new name
    .names = "z_{.col}")) %>%
  mutate(status = "US Native",
         RID = as.character(RID))

normative_profile_trait = immigrants_profile_trait %>%
  full_join(us_profile_trait) %>%
  select(trait, z_response) %>%
  filter(!is.na(z_response)) %>%
  group_by(trait) %>%
  summarise(normative = mean(z_response))
```

We re-estimate the model of immigrant PE-fit using trait profiles, rather than item profiles.

```
mod1_trait = immigrants_profile_trait %>%
  full_join(normative_profile_trait) %>%
  lmer(z_response ~ z_zip_response + normative + (-1 + z_zip_response | ZCTA/RID),
            data = .)
```

```
load(here("objects/mod1-trait.Rdata"))
```

```
summary(mod1_trait)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite's method [
## lmerModLmerTest]
## Formula: z_response ~ z_zip_response + normative + (-1 + z_zip_response |
##     ZCTA/RID)
##    Data: .
##
## REML criterion at convergence: 1404681
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -3.7854 -0.7109  0.0155  0.7292  4.0021
##
## Random effects:
##  Groups   Name           Variance Std.Dev.
##  RID:ZCTA z_zip_response 0.034103 0.1847
##  ZCTA     z_zip_response 0.008209 0.0906
##  Residual                0.899533 0.9484
## Number of obs: 509022, groups:  RID:ZCTA, 19580; ZCTA, 2081
##
## Fixed effects:
##                        Estimate       Std. Error                df t value
## (Intercept)         0.0000002244      0.0013293529 489440.5806289451    0.00
## z_zip_response      0.0604099857      0.0035460130   1993.8771741957   17.04
## normative          -0.1693282054      0.0145138694 488214.6008305461  -11.67
##                              Pr(>|t|)
## (Intercept)                         1
## z_zip_response <0.0000000000000002 ***
## normative      <0.0000000000000002 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) z_zp_r
## z_zip_rspns  0.000
## normative    0.000 -0.176
```

5.2 Restricting analyses to only participants with complete data

Here we fit Model 1 using items and present the results with the original model.

```
mod1_item_full = immigrants_profile %>%
  with_groups(RID, mutate, n = n()) %>%
```

```
  filter(n == 130) %>%
  full_join(normative_profile) %>%
  lmer(z_response ~ z_zip_response + normative + (-1 + z_zip_response | ZCTA/RID),
          data = .)
```

```
summary(mod1_item_full)
```

```
## Linear mixed model fit by REML. t-tests use Satterthwaite’s method [
## lmerModLmerTest]
## Formula: z_response ~ z_zip_response + normative + (-1 + z_zip_response |
##     ZCTA/RID)
##    Data: .
##
## REML criterion at convergence: 1688255
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -5.1595 -0.7471  0.1009  0.7552  4.5722
##
## Random effects:
##  Groups   Name           Variance Std.Dev.
##  RID:ZCTA z_zip_response 0.020025 0.14151
##  ZCTA     z_zip_response 0.004418 0.06647
##  Residual                0.782471 0.88457
## Number of obs: 648180, groups:  RID:ZCTA, 4986; ZCTA, 1450
##
## Fixed effects:
##                      Estimate    Std. Error            df t value
## (Intercept)          0.001414      0.001099 643192.215370   1.287
## z_zip_response       0.237777      0.004381   3143.384668  54.280
## normative            0.463120      0.006191 643773.005697  74.807
##                            Pr(>|t|)
## (Intercept)                   0.198
## z_zip_response <0.0000000000000002 ***
## normative      <0.0000000000000002 ***
## ---
## Signif. codes:  0 ’***’ 0.001 ’**’ 0.01 ’*’ 0.05 ’.’ 0.1 ’ ’ 1
##
## Correlation of Fixed Effects:
##             (Intr) z_zp_r
## z_zip_rspns -0.011
## normative    0.017 -0.630
```

5.3 PE-fit based on correlations

Other PE-fit work calculates person-level correlations and applies the Fisher r-to-z transformation when predicting fit from other variables. We believe this approach is flawed in this specific instance, given that participants may have provided different amounts of information. We calculate fit from this method here. We use both the item-level profiles and trait-level profiles for completeness.

Table S25: Descriptive statistics for different methods of calculating PE-fit.

	Multilevel modeling		Correlation	
Statistic	Item profile	Trait profile	Item profile	Trait profile
mean	0.23	0.06	0.38	0.11
median	0.24	0.07	0.40	0.12
sd	0.12	0.13	0.19	0.30
min	-0.53	-0.43	-0.58	-0.77
max	0.60	0.49	0.86	0.86

Table S26: Correlations between different methods of calcuating PE-fit

	mlm_item	mlm_trait	cor_item	cor_trait
mlm_item	1.00		0.95	0.35
mlm_trait		1.00	0.39	0.91
cor_item	0.95	0.39	1.00	0.34
cor_trait	0.35	0.91	0.34	1.00

```r
# PE_fit using items
item_fit = immigrants_profile %>%
  with_groups(RID, nest) %>%
  mutate(cor_item = map_dbl(data,
                            ~cor(.$z_response, .$z_zip_response,
                                 use = "pairwise")),
         n_items = map_dbl(data, nrow)) %>%
  select(-data)

# PE_fit using traits
trait_fit = immigrants_profile_trait %>%
  with_groups(RID, nest) %>%
  mutate(cor_trait = map_dbl(data,
                             ~cor(.$z_response, .$z_zip_response,
                                  use = "pairwise")),
         n_traits = map_dbl(data, nrow)) %>%
  select(-data)
```

5.4 Compare all methods

To compare different methods of calculating PE-fit, we report here the central tendencies (means and medians) and spread (sd and range) of each method.

These values are presented in Table S25.

We also extract estimates for each immigrant and calculate the correlations between them. When using correlations, we transform these using the Fisher r-to-z transformation.

These are shown in Table S26

Finally, we may be interested to know whether some of our correlates are differentialy associated with different measures of calculating PE-fit. Rather than refit all the models here, we'll just quickly correlate our estimates with several numeric predictor variables.

Table S27: Correlation of methods with numeric predictors

	mlm_item	mlm_trait	cor_item	cor_trait
prop_foreign	0.00		0.04	0.06
prop_originContinent	-0.03		0.00	-0.02
prop_ancestry	0.02		0.07	0.02
match_originCountry	0.03		0.07	-0.05
SPI_Extra	0.09		0.11	-0.12
SPI_Agree	0.48		0.50	-0.02
SPI_Consc	0.40		0.42	-0.10
SPI_Neuro	-0.14		-0.19	0.13
SPI_Open	0.40		0.40	0.03

These are shown in Table S27.

5.5 Using all participants regardless of English fluency

Here we fit Model 1 using items and present the results with the original model.

```
load(here("data/immigrants-all.Rdata"))

zip_profiles = us_data %>%
  group_by(ZCTA) %>%
  summarise(
    across( starts_with("q_"), mean, na.rm=T), # mean of each personality item
    n_zip = n() # number of participants
  ) %>%
  ungroup() %>%
  filter(n_zip >= 30)


zip_profile_long = zip_profiles %>%
  #transform to long form
  pivot_longer(
    # name of column storing variable names
    names_to = "item",
    # name of column storing values
    values_to = "zip_response",
    # which variables to stack
    cols = starts_with("q_"))

immigrants_profile_all = immigrants_all %>%
  gather(item, response, starts_with("q_")) %>%
  filter(!is.na(response)) %>%
  #merge with location profiles
  left_join(zip_profile_long) %>%
  filter(!is.na(zip_response)) %>%
  # standardize responses within profile
  with_groups(RID, mutate, across(
    # mutate these columns
    contains("response"),
    # with the z-score function
    z_fun,
    # create new name
    .names = "z_{.col}")) %>%
  mutate(status = "Immigrant",
         RID = as.character(RID))

mod1_all_immigrants = immigrants_profile_all %>%
  with_groups(RID, mutate, n = n()) %>%
  filter(n >= 50) %>%
  full_join(normative_profile) %>%
  lmer(z_response ~ z_zip_response + normative + (-1 + z_zip_response | ZCTA/RID),
          data = .)

mod1_english = immigrants_profile_all %>%
  with_groups(RID, mutate, n = n()) %>%
  filter(n >= 50) %>%
  full_join(normative_profile) %>%
```

```
  lmer(z_response ~ z_zip_response*english + normative + (-1 + z_zip_response | ZCTA/RID),
            data = .)
```

```
broom::tidy(mod1_all_immigrants, conf.int = T, conf.level = .95) %>%
  mutate(
    p.value = printp(p.value),
    across(where(is.numeric), printnum)
  ) %>%
  select(term, group, estimate, std.error, p.value, contains("conf"))
```

```
## # A tibble: 6 x 7
##   term               group    estimate std.error p.value  conf.low conf.high
##   <chr>              <chr>    <chr>    <chr>     <chr>    <chr>    <chr>
## 1 (Intercept)        <NA>     0.00     0.00      ".558"   0.00     0.00
## 2 z_zip_response     <NA>     0.23     0.00      "< .001" 0.22     0.23
## 3 normative          <NA>     0.40     0.00      "< .001" 0.40     0.41
## 4 sd__z_zip_response RID:ZCTA 0.15     NA        ""       NA       NA
## 5 sd__z_zip_response ZCTA     0.06     NA        ""       NA       NA
## 6 sd__Observation    Residual 0.90     NA        ""       NA       NA
```

```
coef(mod1_all_immigrants)$`RID:ZCTA` %>%
  summarise(min = min(z_zip_response),
            max = max(z_zip_response))
```

```
##          min      max
## 1 -0.5333314 0.613654
```

```
emtrends(mod1_english,
         pairwise~english,
         var = "z_zip_response",
         adjust = "holm",
         lmerTest.limit = 2130793)
```

```
## $emtrends
##  english                 z_zip_response.trend      SE    df lower.CL upper.CL
##  NotAtAllNeedTranslation                0.117 0.01063 22772    0.096    0.138
##  NotWell                                0.123 0.00572 20511    0.111    0.134
##  Well                                   0.187 0.00312  6001    0.180    0.193
##  VeryWellFluentNative                   0.264 0.00284  4865    0.259    0.270
##
## Degrees-of-freedom method: satterthwaite
## Confidence level used: 0.95
##
## $contrasts
##  contrast                                       estimate      SE    df t.ratio
##  NotAtAllNeedTranslation - NotWell              -0.00573 0.01150 21895  -0.498
##  NotAtAllNeedTranslation - Well                 -0.06964 0.01051 21967  -6.628
##  NotAtAllNeedTranslation - VeryWellFluentNative -0.14759 0.01052 21955 -14.032
##  NotWell - Well                                 -0.06391 0.00541 21515 -11.813
##  NotWell - VeryWellFluentNative                 -0.14186 0.00545 21599 -26.025
##  Well - VeryWellFluentNative                    -0.07795 0.00260 20971 -30.035
```

```
##  p.value
##   0.6184
##   <.0001
##   <.0001
##   <.0001
##   <.0001
##   <.0001
##
## Degrees-of-freedom method: satterthwaite
## P value adjustment: holm method for 6 tests
```

```
immigrants_all %>% count(english)
```

```
##                    english     n
## 1 NotAtAllNeedTranslation  1086
## 2                  NotWell  2983
## 3                     Well 19632
## 4    VeryWellFluentNative 22230
## 5                     <NA>   864
```

6 References

Bleidorn, Wiebke, Felix Schönbrodt, Jochen E. Gebauer, Peter J. Rentfrow, Jeff Potter, and Samuel D. Gosling. 2016. "To Live Among Like-Minded Others." *Psychological Science* 27 (3): 419–27. https://doi.org/10.1177/0956797615627133.

Church, A Timothy, Jose Alberto S Reyes, Marcia S Katigbak, and Stephanie D Grimm. 1997. "Filipino Personality Structure and the Big Five Model: A Lexical Approach." *Journal of Personality* 65 (3): 477–528.

Condon, David M. 2017. "The SAPA Personality Inventory: An Empirically Derived, Hierarchically Organized Self Report Personality Assessment Model." *PsyArXiv*. https://doi.org/10.31234/osf.io/sc4p9.

Cutler, Andrew, and David M. Condon. 2022. "Deep Lexical Hypothesis: Identifying Personality Structure in Natural Language." *Journal of Personality and Social Psychology*, November. https://doi.org/10.1037/pspp0000443.

De Raad, Boele, Dick PH Barelds, Eveline Levert, Fritz Ostendorf, Boris Mlačić, Lisa Di Blas, Martina Hřebíčková, et al. 2010. "Only Three Factors of Personality Description Are Fully Replicable Across Languages: A Comparison of 14 Trait Taxonomies." *Journal of Personality and Social Psychology* 98 (1): 160. https://doi.org/10.1037/a0017184.

Denissen, Jaap J. A., Wiebke Bleidorn, Marie Hennecke, Maike Luhmann, Ulrich Orth, Jule Specht, and Julia Zimmermann. 2018. "Uncovering the Power of Personality to Shape Income." *Psychological Science* 29 (1): 3–13. https://doi.org/10.1177/0956797617724435.

Laajaj, Rachid, Karen Macours, Daniel Alejandro Pinzon Hernandez, Omar Arias, Samuel D. Gosling, Jeff Potter, Marta Rubio-Codina, and Renos Vakis. 2019. "Challenges to Capture the Big Five Personality Traits in Non-WEIRD Populations." *Science Advances* 5 (7): eaaw5226.

Laher, Sumaya, and Safia Dockrat. 2019. "The Five-Factor Model and Individualism and Collectivism in South Africa: Implications for Personality Assessment." *African Journal of Psychological Assessment* 1 (1): 1–9.

Ludeke, Steven G., and Erik Gahner Larsen. 2017. "Problems with the Big Five Assessment in the World Values Survey." *Personality and Individual Differences* 112: 103–5.

Ryff, Carol D., and Corey Lee M. Keyes. 1995. "The Structure of Psychological Well-Being Revisited." *Journal of Personality and Social Psychology* 69 (4): 719. https://doi.org/10.1037/0022-3514.69.4.719.

Thalmayer, Amber Gayle, and Gerard Saucier. 2014. "The Questionnaire Big Six in 26 Nations: Developing Cross–culturally Applicable Big Six, Big Five and Big Two Inventories." *European Journal of Personality* 28 (5): 482–96.

Wood, Dustin, and R. Michael Furr. 2016. “The Correlates of Similarity Estimates Are Often Misleadingly Positive: The Nature and Scope of the Problem, and Some Solutions.” *Personality and Social Psychology Review* 20 (2): 79–99. https://doi.org/10.1177/1088868315581119.

A IRT curves for SPI-27 measures

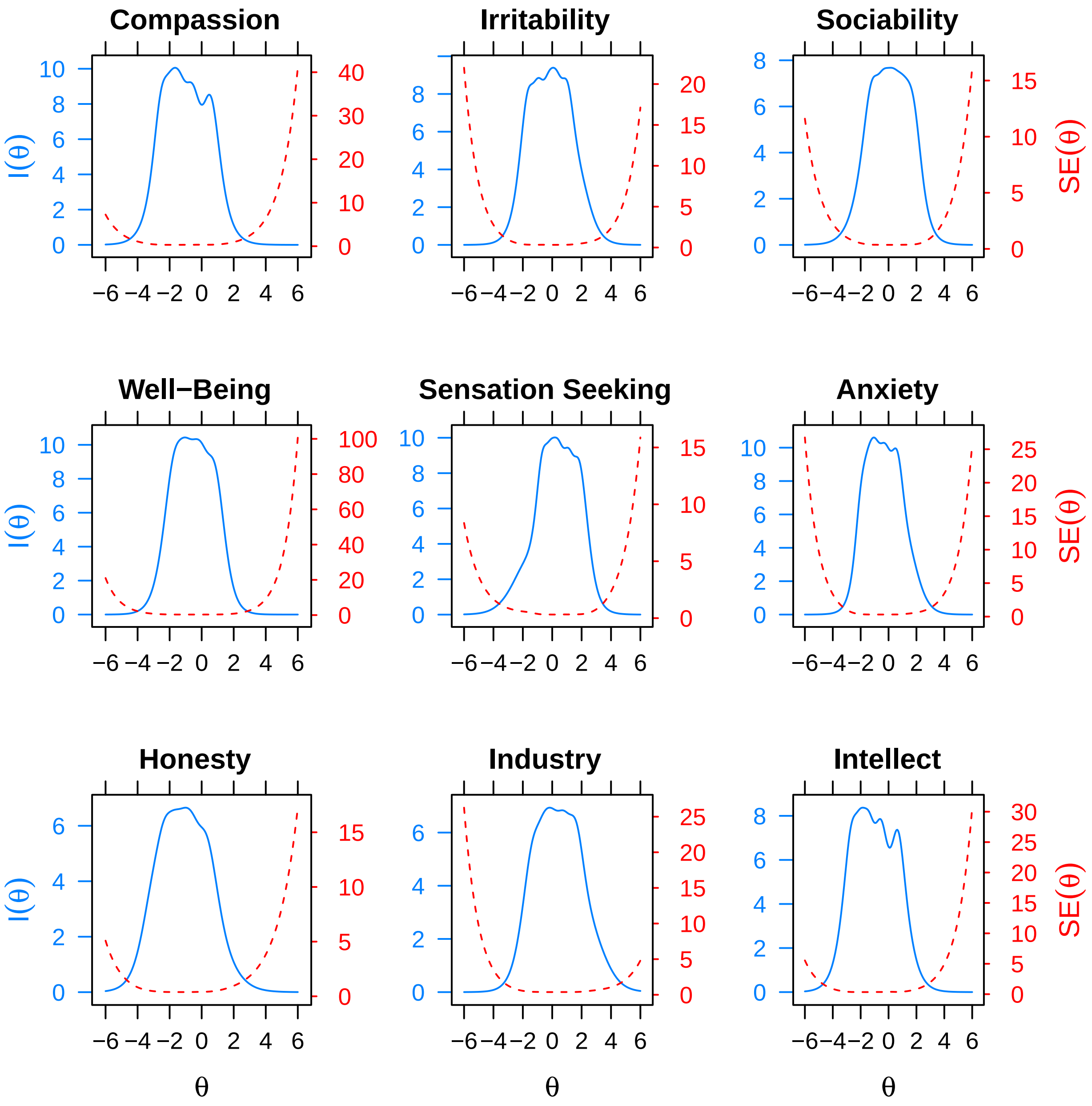


Figure S13: Item information curves for SPI-27 narrow traits: Compassion, Irritability, Sociability, Well-Being, Sensation Seeking, Anxiety, Honesty, Industry, and Intellect. Blue lines represent the informatoin of the measure at each value (standard deviation) of the latent trait, while red lines represent the standard error of the estiamte.

B Response Surface Analysis

Response surface analysis (RSA) is a commonly employed method in the study of person-environment fit because it allows for assessment of mismatch (as well as match), non-linear effects, and complementary congruence. However, we felt that this method did not meet the overall goals of the project, as the method currently does not allow for estimation of how fit is predicted by categorical variable (e.g., country of origin) and also makes assumptions about measurement invariance across outcome or predictor variables. This latter point was especially challenging, given prior work demonstrating poor congruence between personality structures across cultures, especially for non-Western cultures (Church et al. 1997; Cutler and Condon 2022; De Raad et al. 2010; Laajaj et al. 2019; Laher and Dockrat 2019; Ludeke and Larsen 2017; Thalmayer and Saucier 2014).

However, such analyses would situation the current research within a broader literature testing the effects of person-environment fit on significant life outcomes (e.g., Bleidorn et al. 2016; Denissen et al. 2018). Here we use RSA to test the relationship of PE-fit of immigrants to their well-being. We will not include any other analyses in this section.

```r
# load package and cleaned data
library(RSA)
```

```r
load(here("objects/cleaned_data.Rdata"))

# select big five traits of ZCTAs

ZCTA_b5_traits = full_join(immigrants, us_data) %>%
  select(ZCTA, SPI_Agree, SPI_Consc, SPI_Extra, SPI_Neuro, SPI_Open) %>%
  rename_all(~str_replace(., "SPI_", "ZCTA_")) %>%
  group_by(ZCTA) %>%
  summarise(across(where(is.numeric), \(x) mean(x, na.rm=T)))

# merge with immigrant data

immigrants_b5_traits = immigrants %>%
  select(RID, ZCTA, SPI_Agree, SPI_Consc, SPI_Extra, SPI_Neuro, SPI_Open, wellbeing) %>%
  full_join(ZCTA_b5_traits)
```

B.1 Agreeablenesss

```r
rsa_agree <- immigrants_b5_traits %>%
  filter(!is.nan(SPI_Agree)) %>%
  filter(!is.nan(ZCTA_Agree)) %>%
  filter(!is.nan(wellbeing)) %>%
  RSA(wellbeing ~ SPI_Agree * ZCTA_Agree, data=.)

save(rsa_agree, file = here("objects/rsa-agree.Rdata"))
```

```r
getPar(rsa_agree) %>%
  mutate(
    pvalue = printp(pvalue),
    across(where(is.numeric), printnum)) %>%
  select(-label)
```

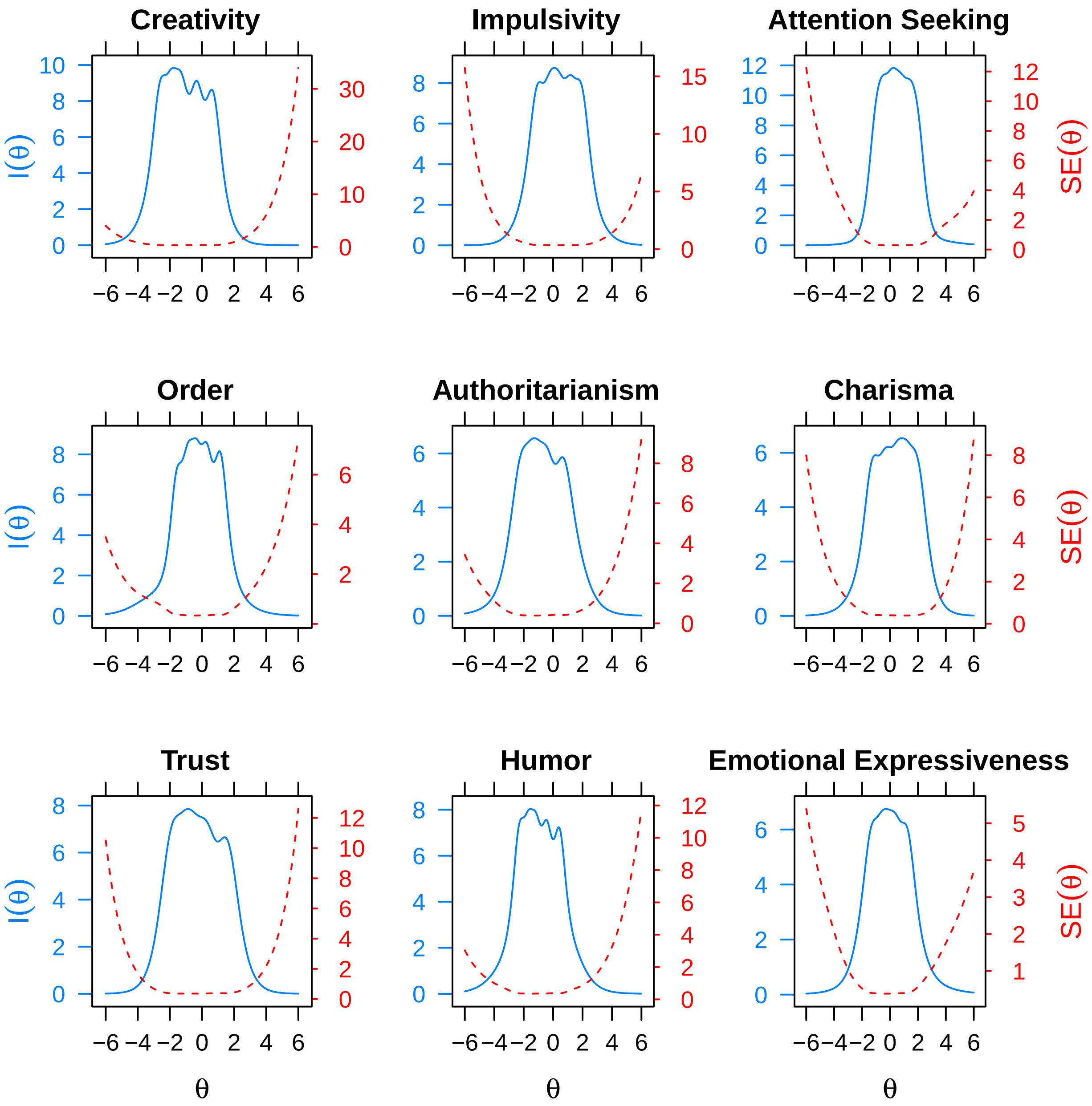


Figure S14: Item information curves for SPI-27 narrow traits: Creativity, Impulsivity, Attention Seeking, Order, Authoritarianism, Charisma, Trust, Humor, and Emotional Expressiveness. Blue lines represent the informatoin of the measure at each value (standard deviation) of the latent trait, while red lines represent the standard error of the estiamte.

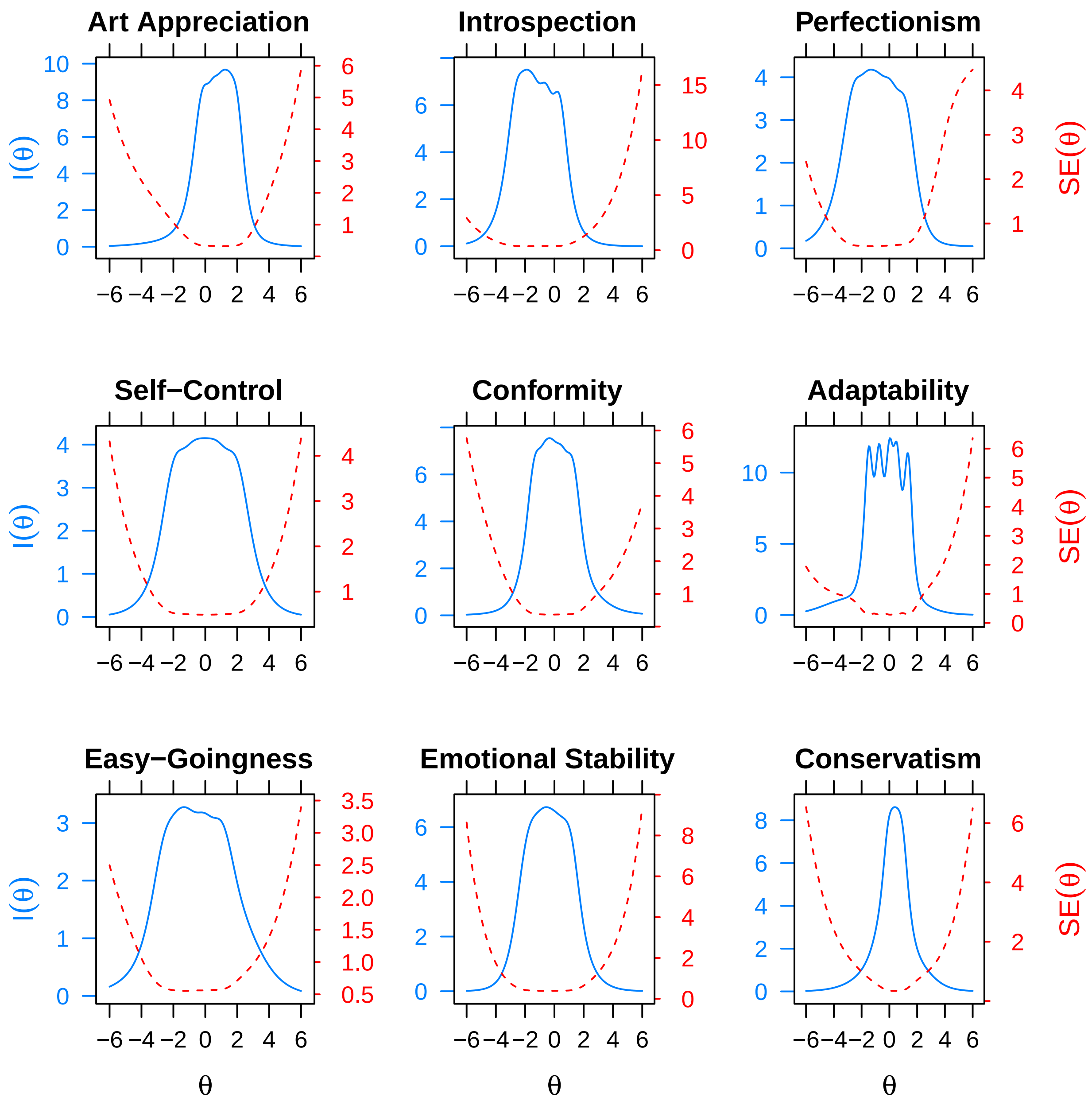


Figure S15: Item information curves for SPI-27 narrow traits: Art Appreciation, Introspection, Perfectionism, Self-Control, Conformity, Adaptability, Easy-Goingness, Emotional Stabiility, and Conservatism. Blue lines represent the informatoin of the measure at each value (standard deviation) of the latent trait, while red lines represent the standard error of the estiamte.

```
##                                                 est   se     z pvalue ci.lower
## wellbeing~SPI_Agree                           -0.32 0.09 -3.57 < .001    -0.49
## wellbeing~ZCTA_Agree                          -0.06 0.12 -0.51   .607    -0.30
## wellbeing~SPI_Agree2                           0.05 0.01  6.89 < .001     0.03
## wellbeing~SPI_Agree_ZCTA_Agree                 0.07 0.03  2.63   .008     0.02
## wellbeing~ZCTA_Agree2                          0.00 0.02 -0.13   .896    -0.05
## wellbeing~1                                   -0.56 0.17 -3.33   .001    -0.89
## a1:=b1+b2                                     -0.38 0.08 -4.68 < .001    -0.54
## a2:=b3+b4+b5                                   0.11 0.01 11.03 < .001     0.09
## a3:=b1-b2                                     -0.26 0.20 -1.31   .191    -0.64
## a4:=b3-b4+b5                                  -0.02 0.05 -0.45   .655    -0.12
## a5:=b3-b5                                      0.05 0.02  2.08   .038     0.00
## X0:=(b2*b4-2*b1*b5)/(4*b3*b5-b4^2)             1.24 0.48  2.62   .009     0.31
## Y0:=(b1*b4-2*b2*b3)/(4*b3*b5-b4^2)             3.08 0.85  3.64 < .001     1.42
## p11:=(b5-b3+sqrt(((b3-b5)^2)+(b4^2)))/b4       0.50 0.09  5.73 < .001     0.33
## p10:=Y0-p11*X0                                 2.46 1.13  2.17   .030     0.24
## p21:=(b5-b3-sqrt((b3-b5)^2+b4^2))/b4          -2.00 0.35 -5.73 < .001    -2.68
## p20:=Y0-p21*X0                                 5.57 0.83  6.68 < .001     3.94
## PA1.curv:=b3+b4*p11+b5*(p11^2)                 0.08 0.01 11.00 < .001     0.06
## PA2.curv:=b3+b4*p21+b5*(p21^2)                -0.10 0.16 -0.63   .532    -0.41
## l1:=(b3+b5+sqrt((b3+b5)^2-4*b3*b5+b4^2))/2     0.06 0.01 12.40 < .001     0.05
## l2:=(b3+b5-sqrt((b3+b5)^2-4*b3*b5+b4^2))/2    -0.02 0.03 -0.66   .506    -0.08
##                                               ci.upper
## wellbeing~SPI_Agree                              -0.14
## wellbeing~ZCTA_Agree                              0.17
## wellbeing~SPI_Agree2                              0.06
## wellbeing~SPI_Agree_ZCTA_Agree                    0.11
## wellbeing~ZCTA_Agree2                             0.05
## wellbeing~1                                      -0.23
## a1:=b1+b2                                        -0.22
## a2:=b3+b4+b5                                      0.13
## a3:=b1-b2                                         0.13
## a4:=b3-b4+b5                                      0.08
## a5:=b3-b5                                         0.10
## X0:=(b2*b4-2*b1*b5)/(4*b3*b5-b4^2)                2.18
## Y0:=(b1*b4-2*b2*b3)/(4*b3*b5-b4^2)                4.75
## p11:=(b5-b3+sqrt(((b3-b5)^2)+(b4^2)))/b4          0.67
## p10:=Y0-p11*X0                                    4.68
## p21:=(b5-b3-sqrt((b3-b5)^2+b4^2))/b4             -1.31
## p20:=Y0-p21*X0                                    7.21
## PA1.curv:=b3+b4*p11+b5*(p11^2)                    0.09
## PA2.curv:=b3+b4*p21+b5*(p21^2)                    0.21
## l1:=(b3+b5+sqrt((b3+b5)^2-4*b3*b5+b4^2))/2        0.07
## l2:=(b3+b5-sqrt((b3+b5)^2-4*b3*b5+b4^2))/2        0.04
```

```
plot(rsa_agree, model="RR",
     points = FALSE,
    xlab="Immigrant Agreeableness",
    ylab="ZCTA Agreeableness",
    zlab="Immigrant Wellbeing")
```

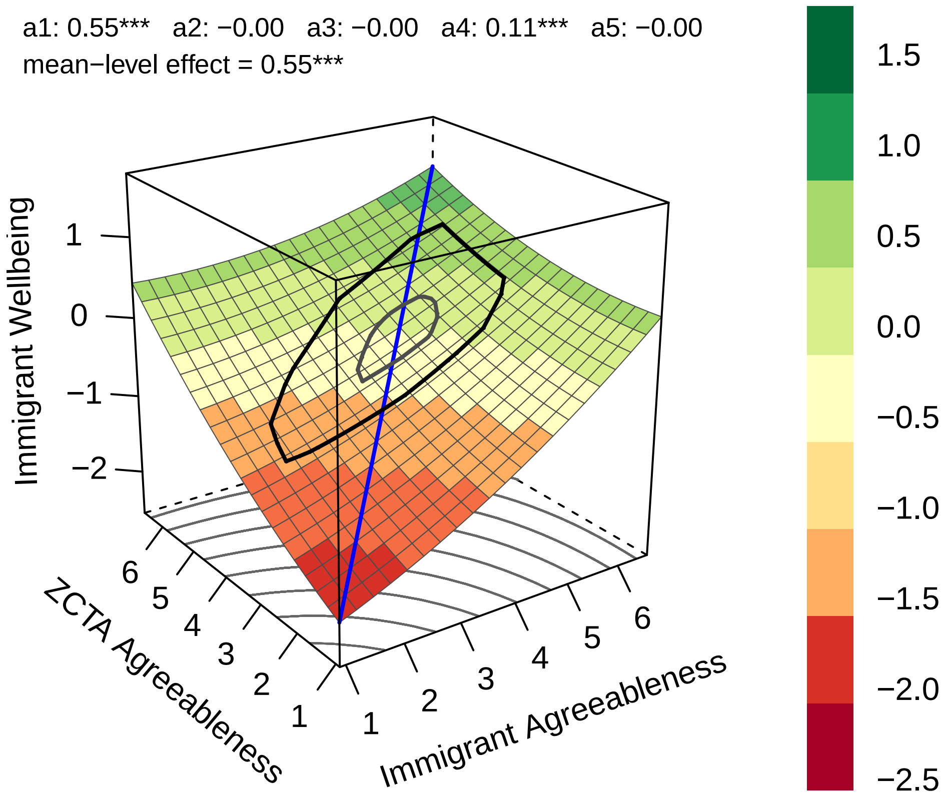


Figure S16: Response surface of RSA analysis for agreeableness and well-being.

```
plot(rsa_agree, model="RR", type = "contour",
     points = FALSE,
    xlab="Immigrant Agreeableness",
    ylab="ZCTA Agreeableness",
    zlab="Immigrant Wellbeing")
```

B.2 Conscientiousness

```
rsa_consc <- immigrants_b5_traits %>%
  filter(!is.nan(SPI_Consc)) %>%
  filter(!is.nan(ZCTA_Consc)) %>%
  filter(!is.nan(wellbeing)) %>%
  RSA(wellbeing ~ SPI_Consc * ZCTA_Consc, data=.)

save(rsa_consc, file = here("objects/rsa-consc.Rdata"))
```

```
getPar(rsa_consc) %>%
  mutate(
    pvalue = printp(pvalue),
    across(where(is.numeric), printnum)) %>%
  select(-label)
```

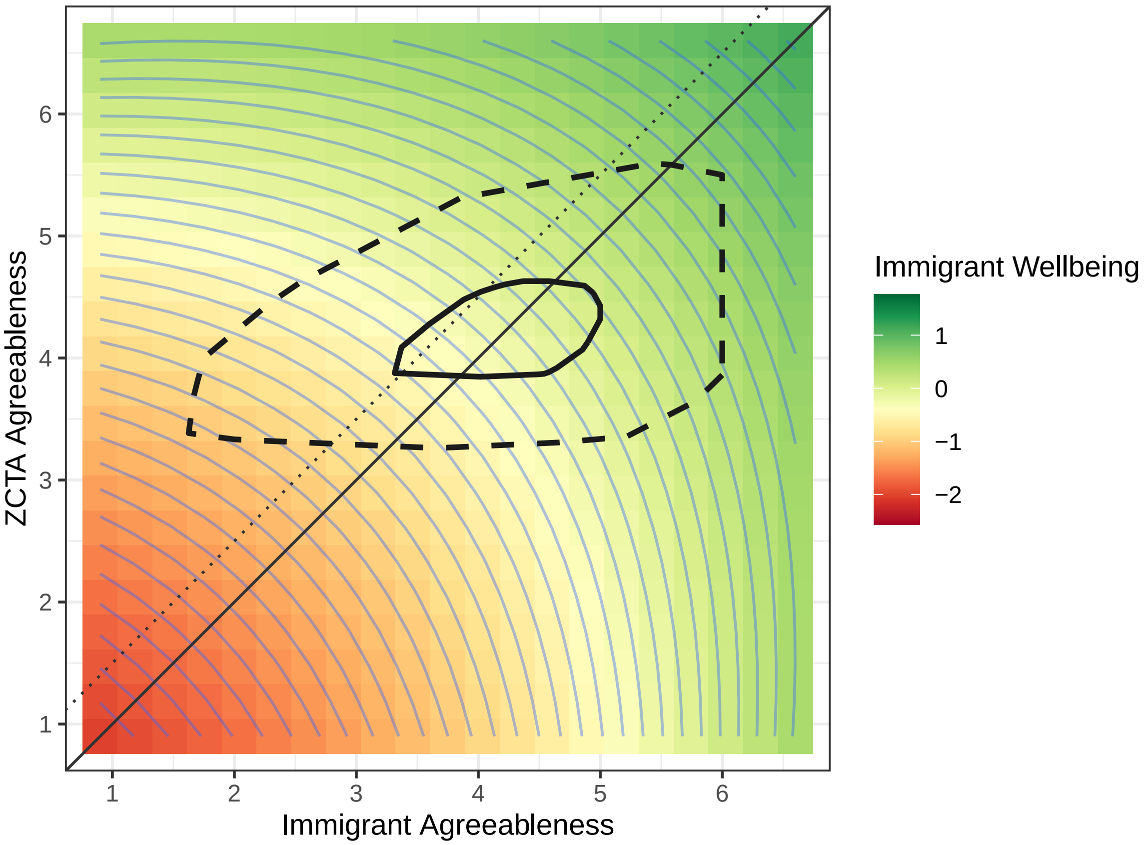


Figure S17: Contour plot of RSA analysis for agreeableness and well-being.

```
##                                              est   se      z pvalue ci.lower
## wellbeing~SPI_Consc                         0.17 0.08   2.28   .023     0.02
## wellbeing~ZCTA_Consc                       -0.08 0.10  -0.78   .437    -0.28
## wellbeing~SPI_Consc2                        0.03 0.01   4.73 < .001     0.02
## wellbeing~SPI_Consc_ZCTA_Consc              0.01 0.02   0.67   .506    -0.03
## wellbeing~ZCTA_Consc2                       0.02 0.02   0.79   .430    -0.03
## wellbeing~1                                -1.56 0.15 -10.64 < .001    -1.84
## a1:=b1+b2                                   0.09 0.07   1.28   .201    -0.05
## a2:=b3+b4+b5                                0.06 0.01   6.59 < .001     0.04
## a3:=b1-b2                                   0.25 0.17   1.53   .127    -0.07
## a4:=b3-b4+b5                                0.03 0.04   0.68   .497    -0.06
## a5:=b3-b5                                   0.01 0.02   0.52   .601    -0.03
## X0:=(b2*b4-2*b1*b5)/(4*b3*b5-b4^2)         -4.18 3.08  -1.36   .174   -10.21
## Y0:=(b1*b4-2*b2*b3)/(4*b3*b5-b4^2)          4.13 6.51   0.63   .526    -8.63
## p11:=(b5-b3+sqrt(((b3-b5)^2)+(b4^2)))/b4    0.50 0.35   1.42   .154    -0.19
## p10:=Y0-p11*X0                              6.22 8.22   0.76   .449    -9.88
## p21:=(b5-b3-sqrt((b3-b5)^2+b4^2))/b4       -2.00 1.40  -1.42   .154    -4.75
## p20:=Y0-p21*X0                             -4.23 4.54  -0.93   .351   -13.12
## PA1.curv:=b3+b4*p11+b5*(p11^2)              0.04 0.01   3.55 < .001     0.02
## PA2.curv:=b3+b4*p21+b5*(p21^2)              0.07 0.13   0.51   .609    -0.19
## l1:=(b3+b5+sqrt((b3+b5)^2-4*b3*b5+b4^2))/2  0.03 0.00   6.92 < .001     0.02
## l2:=(b3+b5-sqrt((b3+b5)^2-4*b3*b5+b4^2))/2  0.01 0.03   0.52   .603    -0.04
##                                            ci.upper
## wellbeing~SPI_Consc                            0.32
## wellbeing~ZCTA_Consc                           0.12
## wellbeing~SPI_Consc2                           0.04
## wellbeing~SPI_Consc_ZCTA_Consc                 0.06
## wellbeing~ZCTA_Consc2                          0.06
## wellbeing~1                                   -1.27
## a1:=b1+b2                                      0.23
## a2:=b3+b4+b5                                   0.08
## a3:=b1-b2                                      0.58
## a4:=b3-b4+b5                                   0.12
## a5:=b3-b5                                      0.05
## X0:=(b2*b4-2*b1*b5)/(4*b3*b5-b4^2)             1.85
## Y0:=(b1*b4-2*b2*b3)/(4*b3*b5-b4^2)            16.90
## p11:=(b5-b3+sqrt(((b3-b5)^2)+(b4^2)))/b4       1.19
## p10:=Y0-p11*X0                                22.33
## p21:=(b5-b3-sqrt((b3-b5)^2+b4^2))/b4           0.75
## p20:=Y0-p21*X0                                 4.66
## PA1.curv:=b3+b4*p11+b5*(p11^2)                 0.06
## PA2.curv:=b3+b4*p21+b5*(p21^2)                 0.32
## l1:=(b3+b5+sqrt((b3+b5)^2-4*b3*b5+b4^2))/2     0.04
## l2:=(b3+b5-sqrt((b3+b5)^2-4*b3*b5+b4^2))/2     0.06
```

```r
plot(rsa_consc, model="RR",
     points = FALSE,
    xlab="Immigrant Conscientiousness",
    ylab="ZCTA Conscientiousness",
    zlab="Immigrant Wellbeing")
```

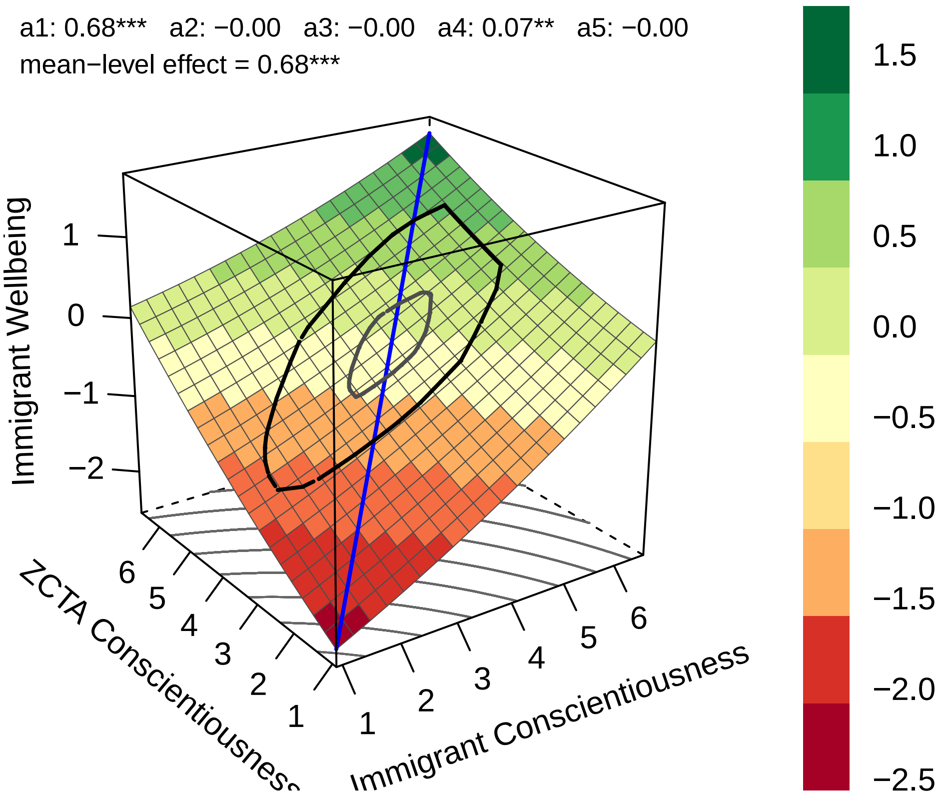


Figure S18: Response surface of RSA analysis for Conscientiousness and well-being.

```
plot(rsa_consc, model="RR", type = "contour",
     points = FALSE,
    xlab="Immigrant Conscientiousness",
    ylab="ZCTA Conscientiousness",
    zlab="Immigrant Wellbeing")
```

B.3 Extraversion

```
rsa_extra <- immigrants_b5_traits %>%
  filter(!is.nan(SPI_Extra)) %>%
  filter(!is.nan(ZCTA_Extra)) %>%
  filter(!is.nan(wellbeing)) %>%
  RSA(wellbeing ~ SPI_Extra * ZCTA_Extra, data=.)

save(rsa_extra, file = here("objects/rsa-extra.Rdata"))
```

```
getPar(rsa_extra) %>%
  mutate(
    pvalue = printp(pvalue),
    across(where(is.numeric), printnum)) %>%
  select(-label)
```

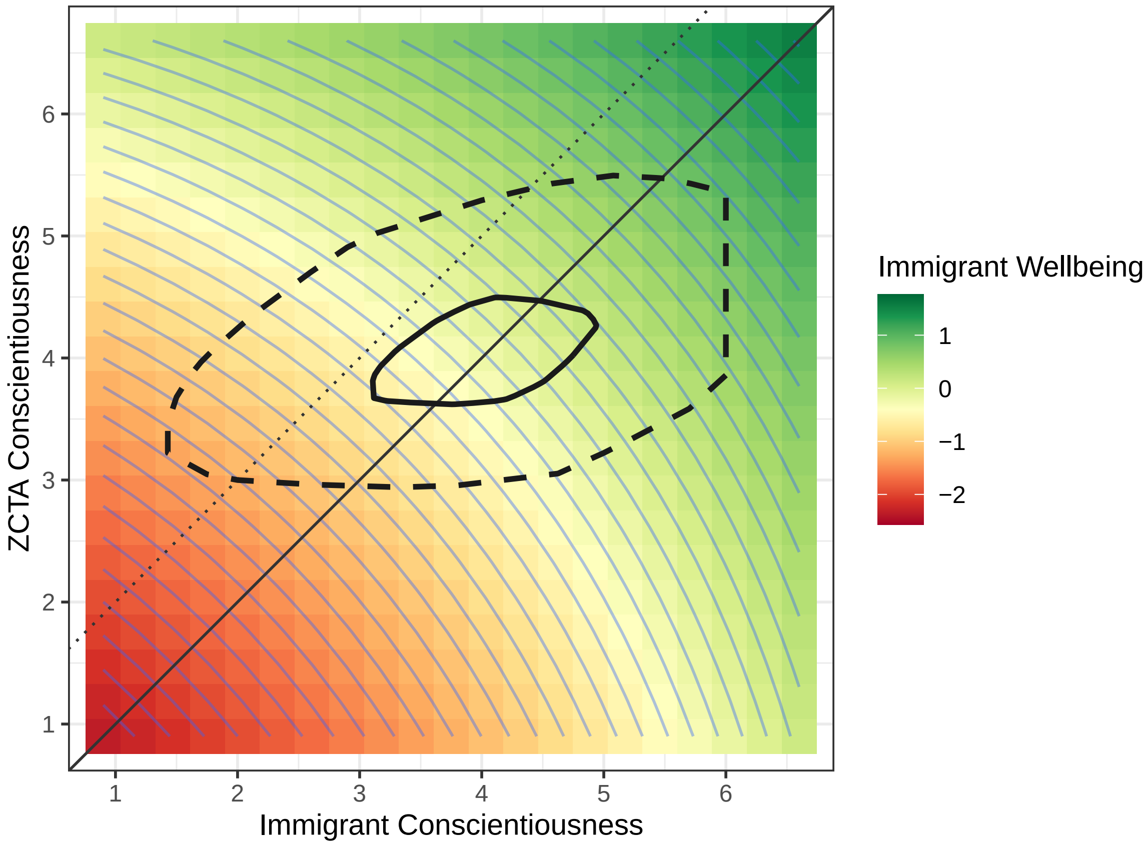


Figure S19: Contour plot of RSA analysis for Conscientiousness and well-being.

```
##                                                est     se       z pvalue
## wellbeing~SPI_Extra                          0.47   0.06   8.23 < .001
## wellbeing~ZCTA_Extra                         0.01   0.08   0.10   .920
## wellbeing~SPI_Extra2                        -0.05   0.00  -9.49 < .001
## wellbeing~SPI_Extra_ZCTA_Extra               0.05   0.02   2.56   .011
## wellbeing~ZCTA_Extra2                       -0.01   0.02  -0.73   .465
## wellbeing~1                                 -1.67   0.09 -17.79 < .001
## a1:=b1+b2                                    0.48   0.05   8.86 < .001
## a2:=b3+b4+b5                                -0.01   0.01  -1.54   .123
## a3:=b1-b2                                    0.46   0.13   3.69 < .001
## a4:=b3-b4+b5                                -0.11   0.04  -2.78   .005
## a5:=b3-b5                                   -0.03   0.02  -1.71   .088
## X0:=(b2*b4-2*b1*b5)/(4*b3*b5-b4^2)          45.88 253.89   0.18   .857
## Y0:=(b1*b4-2*b2*b3)/(4*b3*b5-b4^2)          78.29 507.68   0.15   .877
## p11:=(b5-b3+sqrt(((b3-b5)^2)+(b4^2)))/b4     1.87   0.97   1.93   .054
## p10:=Y0-p11*X0                              -7.59   6.99  -1.09   .277
## p21:=(b5-b3-sqrt((b3-b5)^2+b4^2))/b4        -0.53   0.28  -1.93   .054
## p20:=Y0-p21*X0                             102.79 632.25   0.16   .871
## PA1.curv:=b3+b4*p11+b5*(p11^2)              -0.01   0.03  -0.16   .870
## PA2.curv:=b3+b4*p21+b5*(p21^2)              -0.07   0.03  -2.24   .025
## l1:=(b3+b5+sqrt((b3+b5)^2-4*b3*b5+b4^2))/2   0.00   0.01  -0.15   .883
## l2:=(b3+b5-sqrt((b3+b5)^2-4*b3*b5+b4^2))/2  -0.06   0.01  -4.34 < .001
##                                              ci.lower ci.upper
## wellbeing~SPI_Extra                             0.36     0.58
## wellbeing~ZCTA_Extra                           -0.14     0.16
## wellbeing~SPI_Extra2                           -0.05    -0.04
## wellbeing~SPI_Extra_ZCTA_Extra                  0.01     0.08
## wellbeing~ZCTA_Extra2                          -0.05     0.02
## wellbeing~1                                    -1.85    -1.48
## a1:=b1+b2                                       0.37     0.58
## a2:=b3+b4+b5                                   -0.03     0.00
## a3:=b1-b2                                       0.22     0.71
## a4:=b3-b4+b5                                   -0.18    -0.03
## a5:=b3-b5                                      -0.07     0.00
## X0:=(b2*b4-2*b1*b5)/(4*b3*b5-b4^2)           -451.75   543.50
## Y0:=(b1*b4-2*b2*b3)/(4*b3*b5-b4^2)           -916.74 1,073.32
## p11:=(b5-b3+sqrt(((b3-b5)^2)+(b4^2)))/b4       -0.03     3.77
## p10:=Y0-p11*X0                                -21.29     6.11
## p21:=(b5-b3-sqrt((b3-b5)^2+b4^2))/b4           -1.08     0.01
## p20:=Y0-p21*X0                             -1,136.39 1,341.97
## PA1.curv:=b3+b4*p11+b5*(p11^2)                 -0.07     0.06
## PA2.curv:=b3+b4*p21+b5*(p21^2)                 -0.14    -0.01
## l1:=(b3+b5+sqrt((b3+b5)^2-4*b3*b5+b4^2))/2     -0.02     0.02
## l2:=(b3+b5-sqrt((b3+b5)^2-4*b3*b5+b4^2))/2     -0.08    -0.03
```

```r
plot(rsa_extra, model="RR",
     points = FALSE,
    xlab="Immigrant Extraversion",
    ylab="ZCTA Extraversion",
    zlab="Immigrant Wellbeing")
```

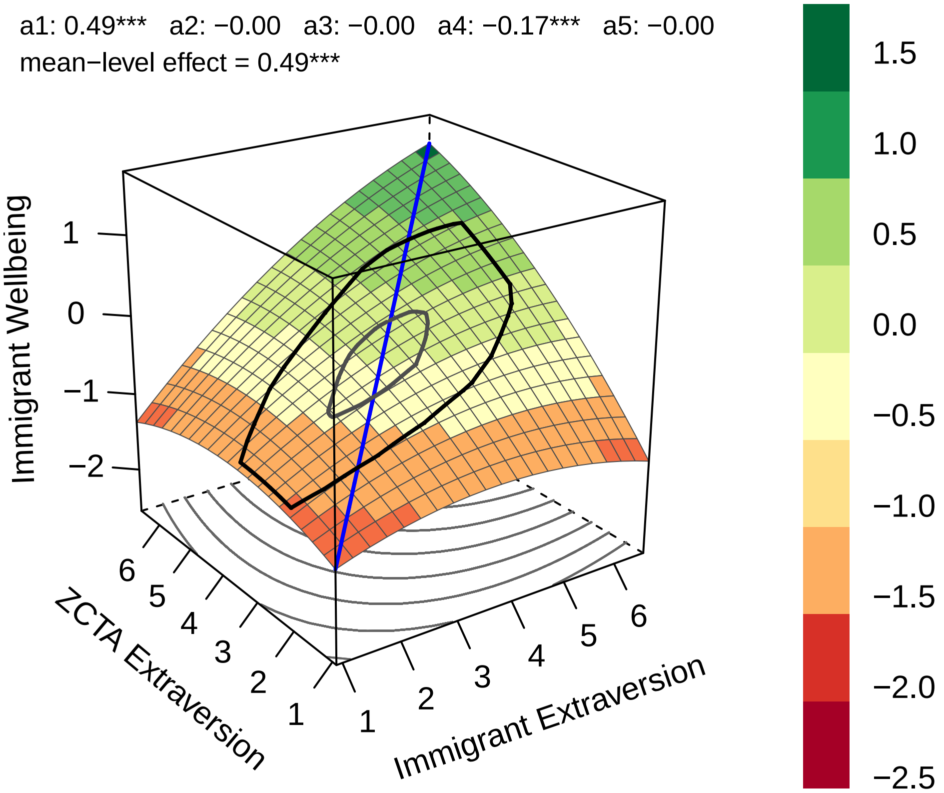


Figure S20: Response surface of RSA analysis for Extraversion and well-being.

```r
plot(rsa_extra, model="RR", type = "contour",
     points = FALSE,
    xlab="Immigrant Extraversion",
    ylab="ZCTA Extraversion",
    zlab="Immigrant Wellbeing")
```

B.4 Neuroticism

```r
rsa_neuro <- immigrants_b5_traits %>%
  filter(!is.nan(SPI_Neuro)) %>%
  filter(!is.nan(ZCTA_Neuro)) %>%
  filter(!is.nan(wellbeing)) %>%
  RSA(wellbeing ~ SPI_Neuro * ZCTA_Neuro, data=.)

save(rsa_neuro, file = here("objects/rsa-neuro.Rdata"))
```

```r
getPar(rsa_neuro) %>%
  mutate(
    pvalue = printp(pvalue),
    across(where(is.numeric), printnum)) %>%
  select(-label)
```

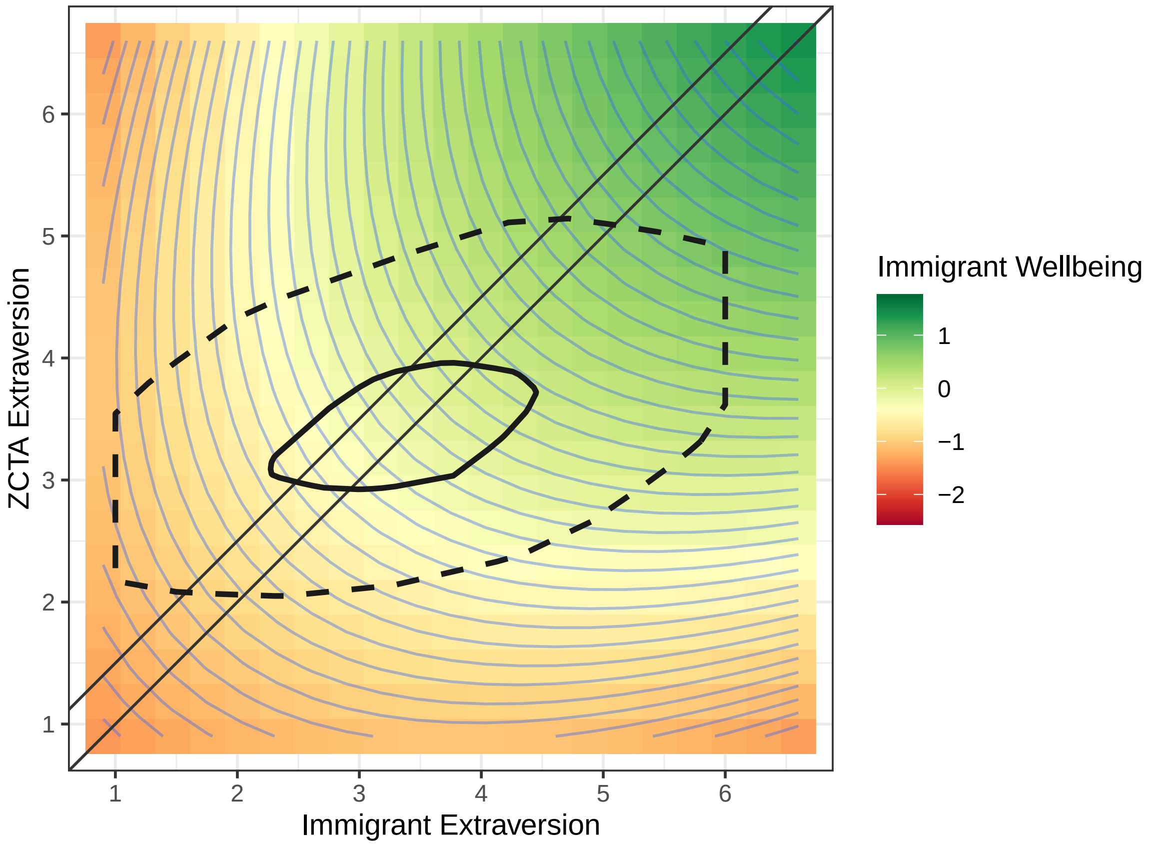


Figure S21: Contour plot of RSA analysis for Extraversion and well-being.

```
##                                               est    se      z pvalue ci.lower
## wellbeing~SPI_Neuro                         -0.31  0.04  -7.82 < .001    -0.39
## wellbeing~ZCTA_Neuro                         0.01  0.06   0.14   .885    -0.11
## wellbeing~SPI_Neuro2                        -0.04  0.00  -9.90 < .001    -0.04
## wellbeing~SPI_Neuro_ZCTA_Neuro              -0.01  0.01  -0.53   .596    -0.03
## wellbeing~ZCTA_Neuro2                       -0.01  0.01  -0.56   .573    -0.03
## wellbeing~1                                  1.80  0.08  22.06 < .001     1.64
## a1:=b1+b2                                   -0.30  0.04  -6.94 < .001    -0.39
## a2:=b3+b4+b5                                -0.05  0.01  -8.33 < .001    -0.06
## a3:=b1-b2                                   -0.32  0.09  -3.52 < .001    -0.50
## a4:=b3-b4+b5                                -0.04  0.02  -1.48   .139    -0.08
## a5:=b3-b5                                   -0.03  0.01  -2.47   .013    -0.05
## X0:=(b2*b4-2*b1*b5)/(4*b3*b5-b4^2)          -4.64  1.09  -4.24 < .001    -6.78
## Y0:=(b1*b4-2*b2*b3)/(4*b3*b5-b4^2)           2.74  5.94   0.46   .645    -8.91
## p11:=(b5-b3+sqrt(((b3-b5)^2)+(b4^2)))/b4    -9.43 14.86  -0.63   .526   -38.56
## p10:=Y0-p11*X0                             -40.99 67.37  -0.61   .543  -173.04
## p21:=(b5-b3-sqrt((b3-b5)^2+b4^2))/b4         0.11  0.17   0.63   .526    -0.22
## p20:=Y0-p21*X0                               3.23  6.81   0.47   .635   -10.11
## PA1.curv:=b3+b4*p11+b5*(p11^2)              -0.58  2.82  -0.21   .837    -6.11
## PA2.curv:=b3+b4*p21+b5*(p21^2)              -0.04  0.00 -12.12 < .001    -0.04
## l1:=(b3+b5+sqrt((b3+b5)^2-4*b3*b5+b4^2))/2  -0.01  0.01  -0.50   .619    -0.03
## l2:=(b3+b5-sqrt((b3+b5)^2-4*b3*b5+b4^2))/2  -0.04  0.00 -11.50 < .001    -0.04
##                                            ci.upper
## wellbeing~SPI_Neuro                           -0.23
## wellbeing~ZCTA_Neuro                           0.12
## wellbeing~SPI_Neuro2                          -0.03
## wellbeing~SPI_Neuro_ZCTA_Neuro                 0.02
## wellbeing~ZCTA_Neuro2                          0.02
## wellbeing~1                                    1.97
## a1:=b1+b2                                     -0.22
## a2:=b3+b4+b5                                  -0.04
## a3:=b1-b2                                     -0.14
## a4:=b3-b4+b5                                   0.01
## a5:=b3-b5                                     -0.01
## X0:=(b2*b4-2*b1*b5)/(4*b3*b5-b4^2)            -2.49
## Y0:=(b1*b4-2*b2*b3)/(4*b3*b5-b4^2)            14.39
## p11:=(b5-b3+sqrt(((b3-b5)^2)+(b4^2)))/b4      19.70
## p10:=Y0-p11*X0                                91.05
## p21:=(b5-b3-sqrt((b3-b5)^2+b4^2))/b4           0.43
## p20:=Y0-p21*X0                                16.57
## PA1.curv:=b3+b4*p11+b5*(p11^2)                 4.95
## PA2.curv:=b3+b4*p21+b5*(p21^2)                -0.03
## l1:=(b3+b5+sqrt((b3+b5)^2-4*b3*b5+b4^2))/2     0.02
## l2:=(b3+b5-sqrt((b3+b5)^2-4*b3*b5+b4^2))/2    -0.03
```

```r
plot(rsa_neuro, model="RR",
     points = FALSE,
    xlab="Immigrant Neuroticism",
    ylab="ZCTA Neuroticism",
    zlab="Immigrant Wellbeing")
```

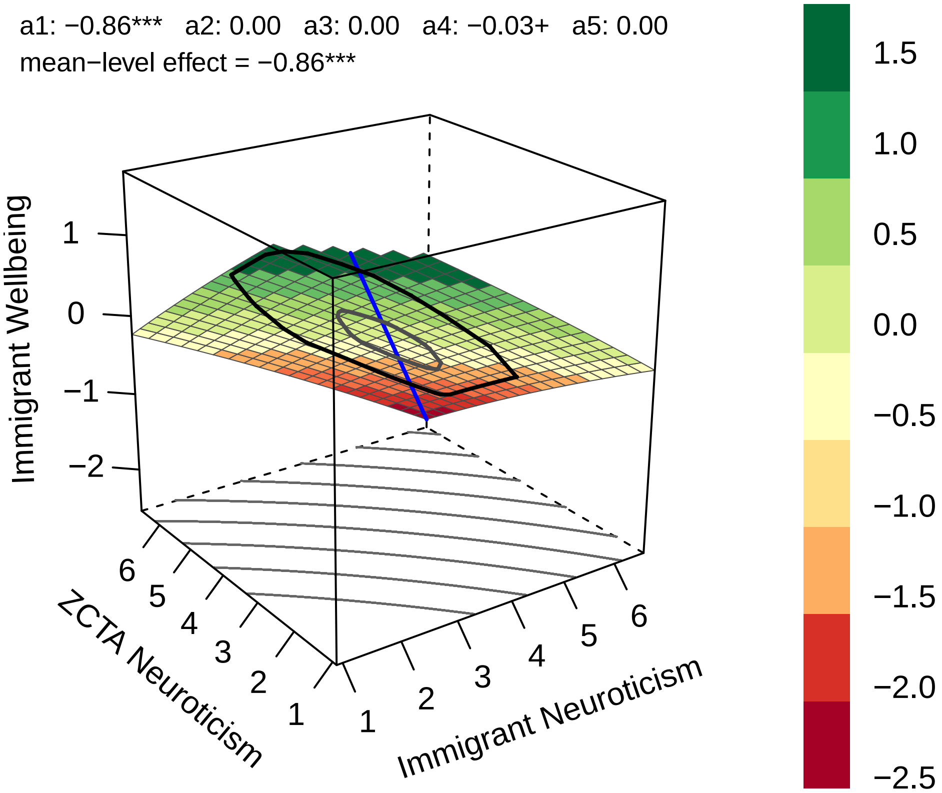


Figure S22: Response surface of RSA analysis for Neuroticism and well-being.

```
plot(rsa_neuro, model="RR", type = "contour",
     points = FALSE,
    xlab="Immigrant Neuroticism",
    ylab="ZCTA Neuroticism",
    zlab="Immigrant Wellbeing")
```

B.5 Openness

```
rsa_open <- immigrants_b5_traits %>%
  filter(!is.nan(SPI_Open)) %>%
  filter(!is.nan(ZCTA_Open)) %>%
  filter(!is.nan(wellbeing)) %>%
  RSA(wellbeing ~ SPI_Open * ZCTA_Open, data=.)

save(rsa_open, file = here("objects/rsa-open.Rdata"))
```

```
getPar(rsa_open) %>%
  mutate(
    pvalue = printp(pvalue),
    across(where(is.numeric), printnum)) %>%
  select(-label)
```

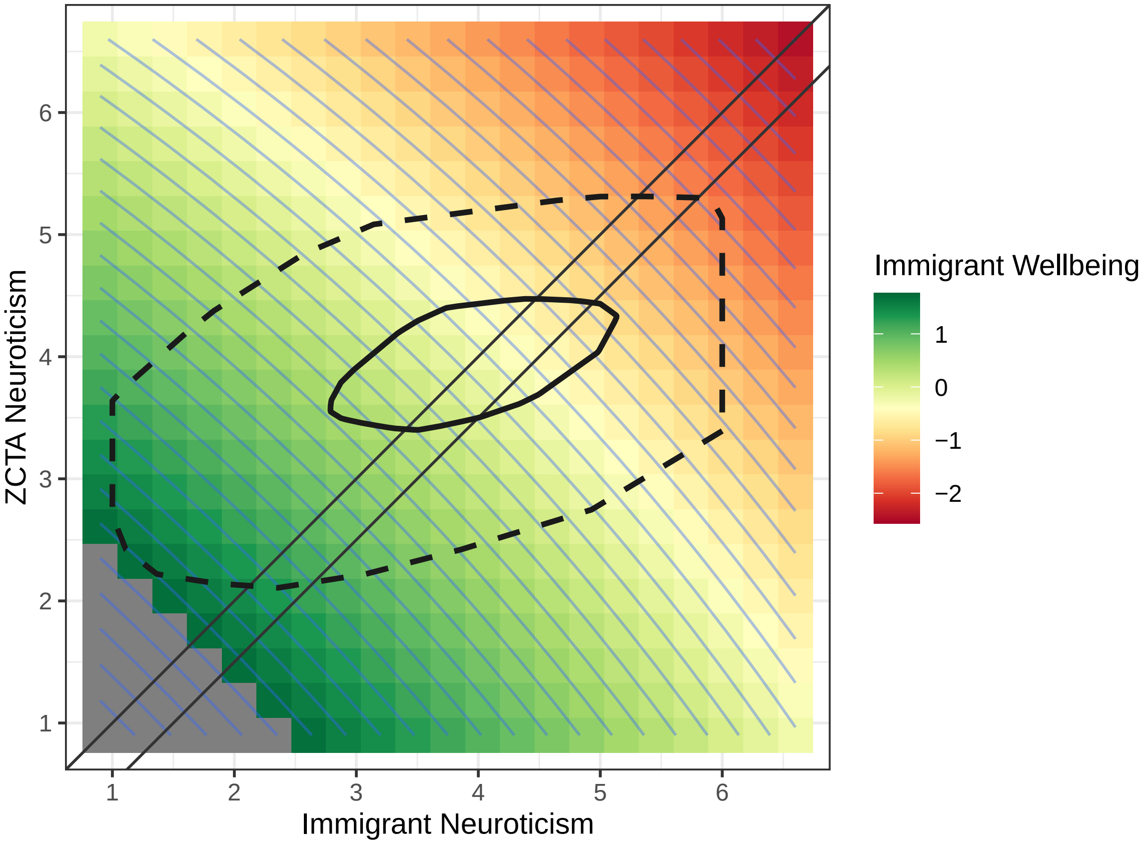


Figure S23: Contour plot of RSA analysis for Neuroticism and well-being.

```
##                                               est    se     z pvalue ci.lower
## wellbeing~SPI_Open                           0.76  0.14  5.60 < .001     0.49
## wellbeing~ZCTA_Open                         -0.32  0.18 -1.76   .078    -0.68
## wellbeing~SPI_Open2                         -0.07  0.01 -7.87 < .001    -0.08
## wellbeing~SPI_Open_ZCTA_Open                 0.03  0.03  0.80   .424    -0.04
## wellbeing~ZCTA_Open2                         0.03  0.03  0.76   .445    -0.04
## wellbeing~1                                 -1.89  0.28 -6.81 < .001    -2.44
## a1:=b1+b2                                    0.44  0.12  3.53 < .001     0.19
## a2:=b3+b4+b5                                -0.01  0.01 -1.00   .318    -0.04
## a3:=b1-b2                                    1.08  0.30  3.63 < .001     0.50
## a4:=b3-b4+b5                                -0.07  0.07 -0.98   .329    -0.20
## a5:=b3-b5                                   -0.09  0.03 -2.80   .005    -0.16
## X0:=(b2*b4-2*b1*b5)/(4*b3*b5-b4^2)           6.33  1.02  6.22 < .001     4.33
## Y0:=(b1*b4-2*b2*b3)/(4*b3*b5-b4^2)           2.99  2.62  1.14   .254    -2.14
## p11:=(b5-b3+sqrt(((b3-b5)^2)+(b4^2)))/b4     7.06 10.54  0.67   .503   -13.59
## p10:=Y0-p11*X0                             -41.71 71.13 -0.59   .558  -181.13
## p21:=(b5-b3-sqrt((b3-b5)^2+b4^2))/b4        -0.14  0.21 -0.67   .503    -0.56
## p20:=Y0-p21*X0                               3.89  1.47  2.64   .008     1.00
## PA1.curv:=b3+b4*p11+b5*(p11^2)               1.41  5.52  0.26   .798    -9.40
## PA2.curv:=b3+b4*p21+b5*(p21^2)              -0.07  0.01 -4.67 < .001    -0.10
## l1:=(b3+b5+sqrt((b3+b5)^2-4*b3*b5+b4^2))/2   0.03  0.03  0.95   .341    -0.03
## l2:=(b3+b5-sqrt((b3+b5)^2-4*b3*b5+b4^2))/2  -0.07  0.01 -5.93 < .001    -0.09
##                                            ci.upper
## wellbeing~SPI_Open                             1.03
## wellbeing~ZCTA_Open                            0.04
## wellbeing~SPI_Open2                           -0.05
## wellbeing~SPI_Open_ZCTA_Open                   0.09
## wellbeing~ZCTA_Open2                           0.09
## wellbeing~1                                   -1.35
## a1:=b1+b2                                      0.68
## a2:=b3+b4+b5                                   0.01
## a3:=b1-b2                                      1.67
## a4:=b3-b4+b5                                   0.07
## a5:=b3-b5                                     -0.03
## X0:=(b2*b4-2*b1*b5)/(4*b3*b5-b4^2)             8.33
## Y0:=(b1*b4-2*b2*b3)/(4*b3*b5-b4^2)             8.12
## p11:=(b5-b3+sqrt(((b3-b5)^2)+(b4^2)))/b4      27.71
## p10:=Y0-p11*X0                                97.70
## p21:=(b5-b3-sqrt((b3-b5)^2+b4^2))/b4           0.27
## p20:=Y0-p21*X0                                 6.77
## PA1.curv:=b3+b4*p11+b5*(p11^2)                12.23
## PA2.curv:=b3+b4*p21+b5*(p21^2)                -0.04
## l1:=(b3+b5+sqrt((b3+b5)^2-4*b3*b5+b4^2))/2     0.08
## l2:=(b3+b5-sqrt((b3+b5)^2-4*b3*b5+b4^2))/2    -0.05
```

```r
plot(rsa_open, model="RR",
     points = FALSE,
    xlab="Immigrant Openness",
    ylab="ZCTA Openness",
    zlab="Immigrant Wellbeing")
```

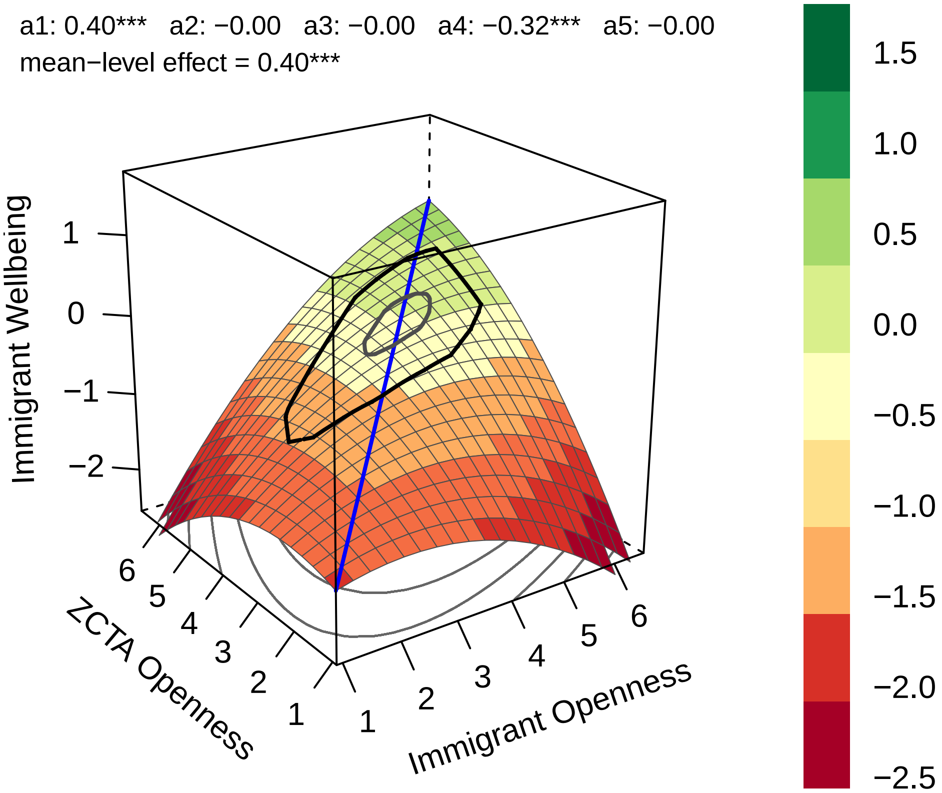


Figure S24: Response surface of RSA analysis for Openness and well-being.

```r
plot(rsa_open, model="RR", type = "contour",
     points = FALSE,
    xlab="Immigrant Openness",
    ylab="ZCTA Openness",
    zlab="Immigrant Wellbeing")
```

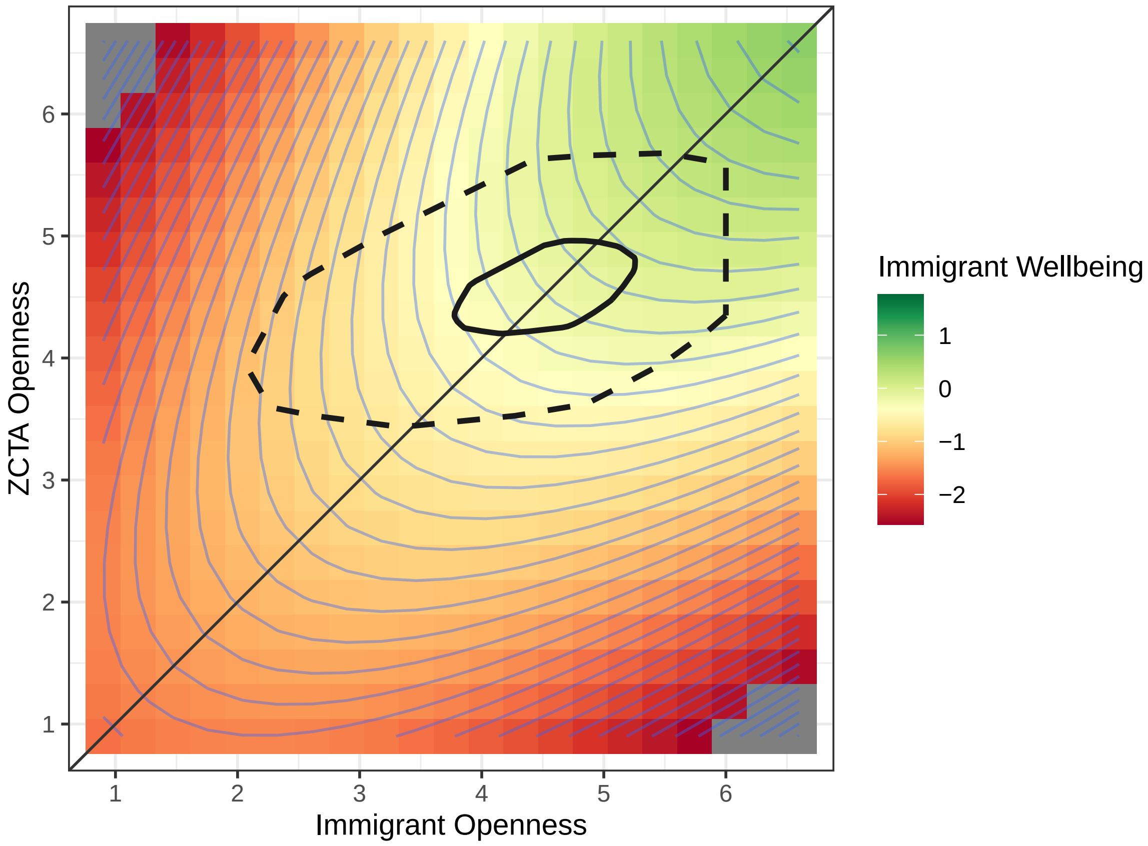


Figure S25: Contour plot of RSA analysis for Openness and well-being.