English proficiency and study-abroad attitudes

August 10, 2017

Contents

Import necessary libraries, read in and clean data

```
library(dplyr)
library(tidyr)
library(ggplot2)
library(lme4)
library(RLRsim)
library(corrplot)
library(stringr)
library(afex)
library(mediation)
library(MASS)
library(scales)
library(car)
library(lsmeans)
library(foreign)
library(reporttools)
library(texreg)
library(PerformanceAnalytics)
setwd('C:/users/ajame/Dropbox/English_stress')
data <- read.csv('All_Data_Original_truncated.csv', stringsAsFactors = FALSE)
```

First we'll rename some of our variables & label the factors. We'll create special versions of the factors for IELTS / TOEFL scores including data from those that took the test only, and versions of factors where levels that are highly underrepresented are eliminated. We'll also create numeric versions of variables that are pseudo-interval / ratio scale.

```
data <-data %>% rename(sex = Q1, urban = Q2, wealth_class = Q3, country = Q4, still_abroad = Q5, durati



data$sex <- factor(data$sex, labels = c("Male", "Female"))

data$urban <- factor(data$urban, labels = c("Country", "Urban"))
```

```
data$wealth_class <- factor(data$wealth_class, labels= c("Lower Class", "Lower-middle class", "Middle cl
data$wealth_class_trimmed <- data$wealth_class
data$wealth_class_trimmed[data$wealth_class_trimmed == "Extremely Wealthy"] <- NA
data$wealth_class_no_extreme <- data$wealth_class
data$wealth_class_no_extreme[data$wealth_class_no_extreme == "Extremely Wealthy"] <- "Upper class"
data$wealth_class_no_extreme <- factor(data$wealth_class_no_extreme, ordered = FALSE)

data$country <- factor(data$country, labels= c("USA", "Canada", "UK / Ireland", "Australia / New Zealand

data$still_abroad <- factor(data$still_abroad, labels= c("Still abroad", "Already returned"))

data$duration_numeric <- dplyr::recode(data$duration, `1`=0.5, `2`=1, `3`=1.5, `4`=2, `5`=3, `6`=4)

data$duration <- factor(data$duration, labels= c("0.5 years", "1 year", "1.5 years", "2 years", "3 Years

data$major <- factor(data$major, labels= c("STEM", "Social sciences", "Business", "Arts", "Languages",

data$took_test <- factor(data$overall_score != 10, labels = c("Did not take test", "Took test"))

data$overall_score_numeric <- dplyr::recode(data$overall_score, `1` = 5, `2` = 5.5, `3` = 6, `4` = 6.5,
data$overall_score_numeric_trimmed <- data$overall_score_numeric
data$overall_score_numeric_trimmed[data$overall_score_numeric_trimmed < 5.5 | data$overall_score_numeric

data$overall_score <- factor(data$overall_score, levels=levels(factor(data$overall_score))[c(11,1:10)],

data$overall_score_takers <- data$overall_score
data$overall_score_takers[data$overall_score_takers == "Did not take"] <- NA
data$overall_score_takers <- factor(data$overall_score_takers)

data$overall_score_trimmed <- data$overall_score
data$overall_score_trimmed[data$overall_score_trimmed %in% c("IELTS 4.5 TOEFL 32-34", "IELTS 5 TOEFL 35-
data$overall_score_trimmed <- factor(data$overall_score_trimmed)

data$overall_score_trimmed_takers <- data$overall_score_trimmed
data$overall_score_trimmed_takers[data$overall_score_trimmed_takers == "Did not take"] <- NA
data$overall_score_trimmed_takers <- factor(data$overall_score_trimmed_takers)

data$took_speaking_test <- factor(data$speaking_score != 10, labels = c("Did not take test", "Took test

data$speaking_score_numeric <- dplyr::recode(data$speaking_score, `1` = 5, `2` = 5.5, `3` = 6, `4` = 6.5
data$speaking_score_numeric_trimmed <- data$speaking_score_numeric
data$speaking_score_numeric_trimmed[data$speaking_score_numeric_trimmed < 5.5 | data$speaking_score_nume

data$speaking_score <- factor(data$speaking_score, levels=levels(factor(data$speaking_score))[c(11,1:10)

data$speaking_score_takers <- data$speaking_score
data$speaking_score_takers[data$speaking_score_takers == "Did not take"] <- NA
data$speaking_score_takers <- factor(data$speaking_score_takers)

data$speaking_score_trimmed <- data$speaking_score
data$speaking_score_trimmed[data$speaking_score_trimmed %in% c("IELTS 4.5 TOEFL 12-13", "IELTS 5 TOEFL 1
data$speaking_score_trimmed <- factor(data$speaking_score_trimmed)
```

```
data$speaking_score_trimmed_takers <- data$speaking_score_trimmed
data$speaking_score_trimmed_takers[data$speaking_score_trimmed_takers == "Did not take"] <- NA
data$speaking_score_trimmed_takers <- factor(data$speaking_score_trimmed_takers)

data$sex_native <- factor(data$sex_native, levels = levels(factor(data$sex_native))[c(2,1)], labels = c
data$sex_chinese <- factor(data$sex_chinese, levels = levels(factor(data$sex_chinese))[c(2,1)], labels =
data$traveled_before <- factor(data$traveled_before, levels = levels(factor(data$traveled_before))[c(2,
```

Demographic data

```
#tableNominal(vars = data %>% dplyr::select(sex, wealth_class, urban, traveled_before, country, duratio
```

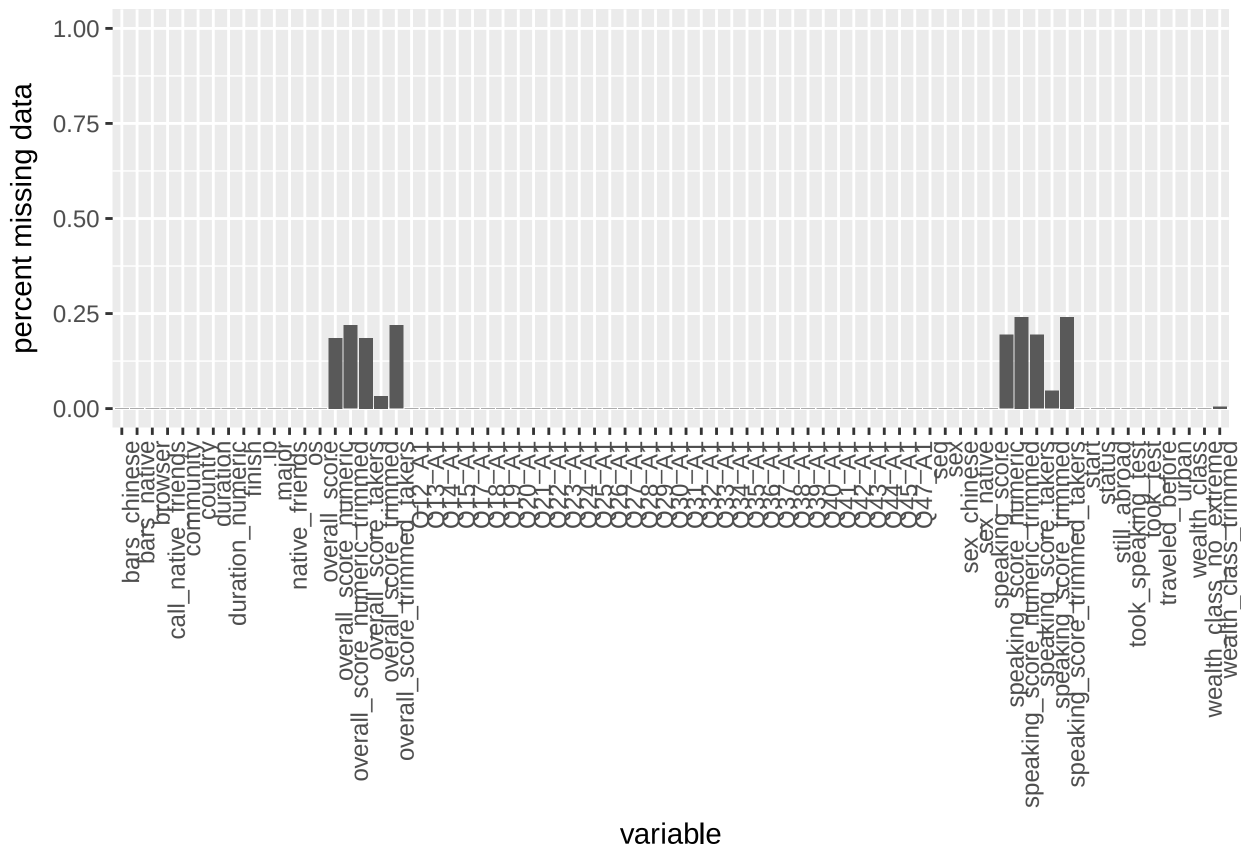
% latex table generated in R 3.4.1 by xtable 1.8-2 package % Sun Sep 17 01:07:16 2017

Variable	**Levels**	**n**	**%**
Gender	Male	118	32.7
	Female	243	67.3
Wealth class	Lower Class	13	3.6
	Lower-middle class	52	14.4
	Middle class	189	52.4
	Upper-middle class	92	25.5
	Upper class	13	3.6
	Extremely Wealthy	2	0.6
Home	Country	64	17.7
	Urban	297	82.3
Prior travel	Never abroad before	251	69.5
	Abroad before	110	30.5
Study abroad country	USA	98	27.1
	Canada	22	6.1
	UK / Ireland	66	18.3
	Australia / New Zealand	73	20.2
	Germany / France / Holland	44	12.2
	Korea / Japan / Singapore	41	11.4
	Other	17	4.7
Duration abroad	0.5 years	48	13.3
	1 year	133	36.8
	1.5 years	39	10.8
	2 years	59	16.3
	3 Years	42	11.6
	4 years or more	40	11.1
Test score	IELTS 4.5 TOEFL 32-34	3	0.8
	IELTS 5 TOEFL 35-45	3	0.8
	IELTS 5.5 TOEFL 46-59	21	5.8
	IELTS 6 TOEFL 60-78	35	9.7
	IELTS 6.5 TOEFL 79-93	70	19.4
	IELTS 7 TOEFL 94-101	80	22.2
	IELTS 7.5 TOEFL 102-109	51	14.1
	IELTS 8 TOEFL 110-114	25	6.9
	IELTS 8.5 TOEFL 115-117	5	1.4
	IELTS 9 TOEFL 118-120	1	0.3
	Did not take	67	18.6
Major	STEM	76	21.1
	Social sciences	44	12.2
	Business	129	35.7
	Arts	34	9.4
	Languages	61	16.9
	Other	17	4.7
	all	361	100.0

Table 1: Demographic characteristics of sample

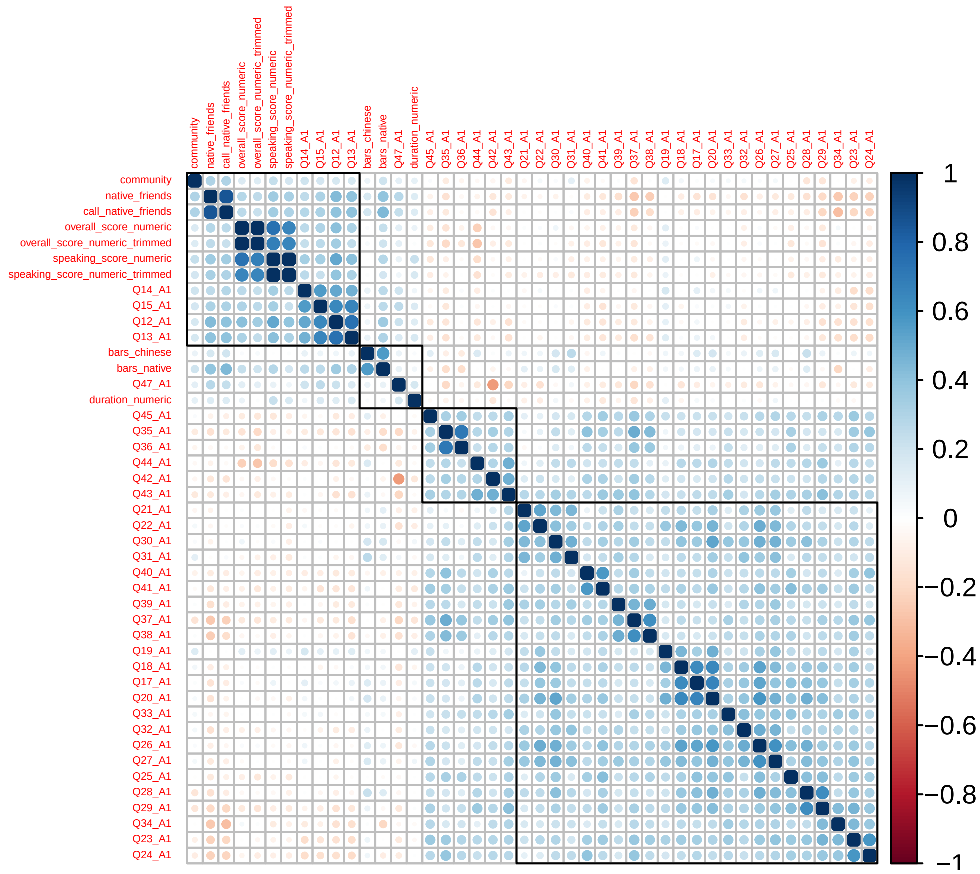
Check for missing data & correlation exploration

```
naPercent <- data %>% summarize_all(funs(sum(is.na(.))/n())) %>% gather(key="variable", value="missing_p
ggplot(naPercent, aes(x=variable, y=missing_percent)) + geom_bar(stat="identity") + theme(axis.text.x =
```



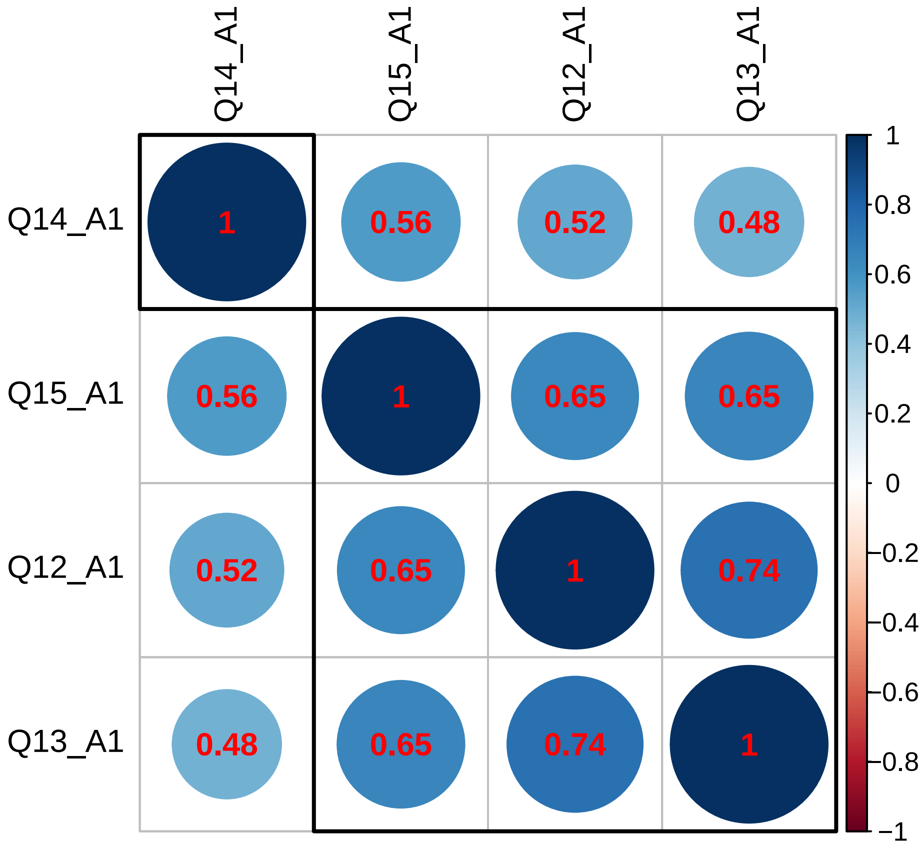
No data is missing except for where we explicitly induced missing data. Let's start converting the appropriate variables to factors.

```
correls <- cor(dplyr::select_if(data, is.numeric) %>% dplyr::select(-seq, -status),  use = "pairwise.com
corrplot(correls, order= "hclust", hclust.method = "complete", addrect = 4, rect.lwd = 1, tl.cex = .3)
```



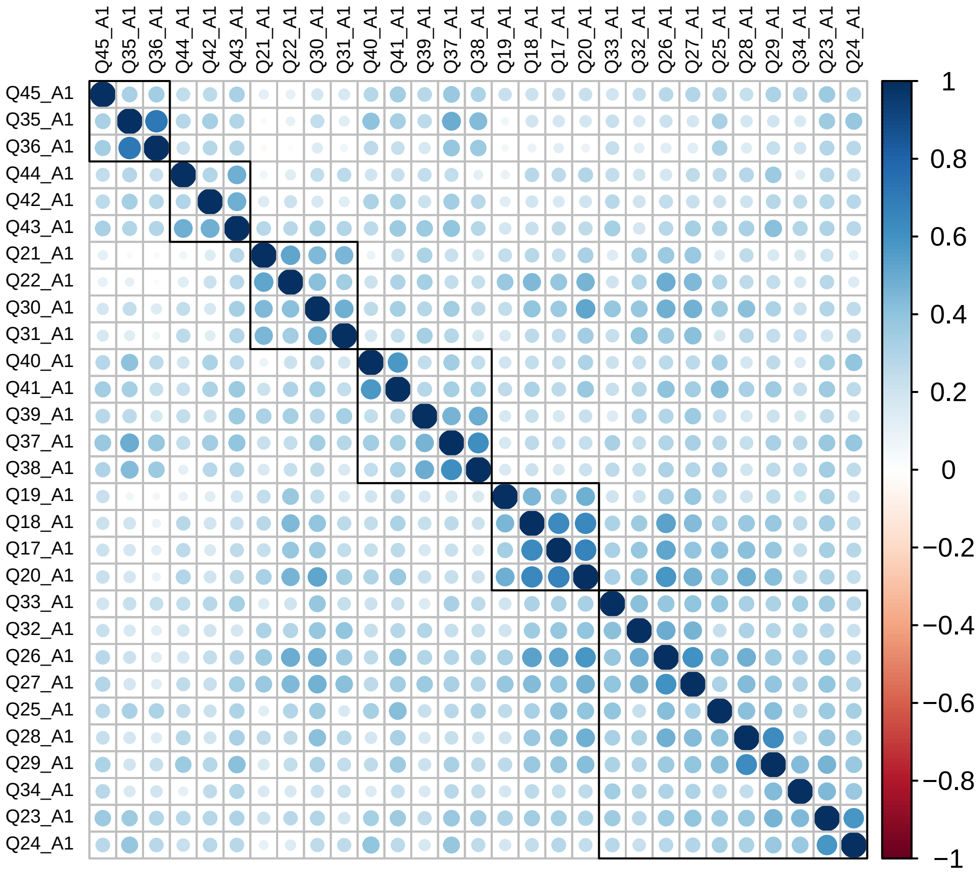
Subsets of variables: our english confidence measures (Q12-Q15) and stress measures (Q17-Q45)

```
correls <- cor(data[,str_c('Q', seq(12, 15), '_A1')])
print(corrplot(correls, order = "hclust", hclust.method = "complete", addrect = 2, rect.lwd = 2, tl.col
```




```
##           Q14_A1    Q15_A1    Q12_A1    Q13_A1
## Q14_A1 1.0000000 0.5620220 0.5185004 0.4776175
## Q15_A1 0.5620220 1.0000000 0.6466257 0.6535074
## Q12_A1 0.5185004 0.6466257 1.0000000 0.7427270
## Q13_A1 0.4776175 0.6535074 0.7427270 1.0000000
```

```
correls <- cor(data[,str_c('Q', seq(17,45), '_A1')])
print(corrplot(correls, order = "hclust", hclust.method = "complete", addrect = 6, rect.lwd = 1, tl.cex
```




```
##            Q45_A1     Q35_A1      Q36_A1     Q44_A1    Q42_A1    Q43_A1
## Q45_A1 1.0000000 0.32852254  0.34240639 0.24875019 0.2667202 0.3261641
## Q35_A1 0.3285225 1.00000000  0.71998966 0.28853075 0.3346569 0.2951666
## Q36_A1 0.3424064 0.71998966  1.00000000 0.22636526 0.2828960 0.2962824
## Q44_A1 0.2487502 0.28853075  0.22636526 1.00000000 0.2929734 0.4824373
## Q42_A1 0.2667202 0.33465688  0.28289595 0.29297341 1.0000000 0.4878840
## Q43_A1 0.3261641 0.29516660  0.29628243 0.48243730 0.4878840 1.0000000
## Q21_A1 0.1172716 0.03359724 -0.03743454 0.06754335 0.1476362 0.2798801
## Q22_A1 0.1097452 0.10108119  0.02597561 0.13965936 0.2114380 0.2718953
## Q30_A1 0.1823539 0.24760428  0.14751350 0.24241037 0.1707464 0.3379615
## Q31_A1 0.1713977 0.13874993  0.06553923 0.26094423 0.1360576 0.2977395
## Q40_A1 0.2823050 0.40383656  0.26182049 0.19483062 0.3165319 0.2642303
## Q41_A1 0.3436279 0.33500277  0.23444761 0.22925701 0.3181222 0.3652849
## Q39_A1 0.2761284 0.26490768  0.17911907 0.24426829 0.2148236 0.3682778
## Q37_A1 0.3769268 0.49264394  0.38141323 0.24772066 0.3440547 0.3910640
## Q38_A1 0.3022819 0.43590495  0.36456826 0.11392232 0.2750687 0.2814172
## Q19_A1 0.2226985 0.06522338  0.05165880 0.09263742 0.1351451 0.1896146
## Q18_A1 0.2149130 0.19917860  0.08809779 0.27117701 0.1993485 0.2275279
## Q17_A1 0.2141231 0.18570067  0.12224375 0.26422233 0.1676001 0.2517736
## Q20_A1 0.2297482 0.18740882  0.08543147 0.29168143 0.2012588 0.2559934
## Q33_A1 0.1981505 0.22870467  0.23783462 0.24819631 0.2748008 0.3378207
## Q32_A1 0.2219153 0.17558086  0.12871088 0.19040544 0.2059927 0.1862129
## Q26_A1 0.2727544 0.21072186  0.13729675 0.18891973 0.2481785 0.2727666
## Q27_A1 0.2994033 0.18699068  0.13216088 0.25009079 0.2295697 0.3372429
## Q25_A1 0.2712351 0.32472338  0.31164969 0.25199513 0.2105711 0.3080282
## Q28_A1 0.2351469 0.18881949  0.14580317 0.28685908 0.2016138 0.3255304
```

```
## Q29_A1 0.3181362 0.19232975  0.23380243 0.36340317 0.2835501 0.4119503
## Q34_A1 0.2765228 0.16746856  0.19088203 0.11017820 0.2502479 0.2958574
## Q23_A1 0.3624822 0.35332245  0.29097941 0.27749682 0.2863765 0.3008900
## Q24_A1 0.2756742 0.38726147  0.27248223 0.22927260 0.2722631 0.2772823
##             Q21_A1     Q22_A1    Q30_A1     Q31_A1     Q40_A1    Q41_A1
## Q45_A1  0.11727163 0.10974522 0.1823539 0.17139772 0.28230502 0.3436279
## Q35_A1  0.03359724 0.10108119 0.2476043 0.13874993 0.40383656 0.3350028
## Q36_A1 -0.03743454 0.02597561 0.1475135 0.06553923 0.26182049 0.2344476
## Q44_A1  0.06754335 0.13965936 0.2424104 0.26094423 0.19483062 0.2292570
## Q42_A1  0.14763623 0.21143800 0.1707464 0.13605757 0.31653194 0.3181222
## Q43_A1  0.27988006 0.27189534 0.3379615 0.29773947 0.26423030 0.3652849
## Q21_A1  1.00000000 0.52533993 0.4407687 0.45805966 0.08418099 0.2173208
## Q22_A1  0.52533993 1.00000000 0.4164842 0.34324538 0.21876131 0.3049912
## Q30_A1  0.44076872 0.41648423 1.0000000 0.48138065 0.25751673 0.3323987
## Q31_A1  0.45805966 0.34324538 0.4813807 1.00000000 0.18632499 0.2474344
## Q40_A1  0.08418099 0.21876131 0.2575167 0.18632499 1.00000000 0.5715669
## Q41_A1  0.21732076 0.30499119 0.3323987 0.24743442 0.57156686 1.0000000
## Q39_A1  0.31471097 0.33251349 0.2874430 0.33735169 0.24268302 0.2868871
## Q37_A1  0.22896118 0.24247668 0.3430241 0.28080687 0.34964736 0.3349697
## Q38_A1  0.16944126 0.23087685 0.2580872 0.16947541 0.24763820 0.3129796
## Q19_A1  0.24115056 0.37631275 0.2477175 0.16010173 0.19068976 0.2551764
## Q18_A1  0.27266044 0.44803036 0.3992267 0.26248942 0.24135794 0.3112292
## Q17_A1  0.23788637 0.38065456 0.3667218 0.24815975 0.23203703 0.2689082
## Q20_A1  0.32625630 0.46007008 0.5296260 0.34197624 0.30462144 0.3732558
## Q33_A1  0.14971766 0.20000626 0.3838486 0.23273156 0.21104440 0.2230266
## Q32_A1  0.31510118 0.29921127 0.3827912 0.39026229 0.23864248 0.2883512
## Q26_A1  0.36952974 0.49735965 0.4887753 0.35340690 0.26275504 0.4009717
## Q27_A1  0.37809142 0.44316450 0.4739480 0.41615919 0.26786583 0.3420537
## Q25_A1  0.13274708 0.29485178 0.3525051 0.16327104 0.33393446 0.4282849
## Q28_A1  0.25504086 0.25196362 0.4107304 0.27550250 0.18967388 0.3203282
## Q29_A1  0.16924627 0.24891044 0.3130594 0.24627846 0.25213863 0.3525574
## Q34_A1  0.16185177 0.17282286 0.2188967 0.21842580 0.15148526 0.2371236
## Q23_A1  0.21271923 0.27003290 0.2986267 0.19277032 0.33551436 0.3558968
## Q24_A1  0.10548450 0.14904914 0.2591695 0.25795543 0.39806767 0.2684365
##           Q39_A1    Q37_A1    Q38_A1     Q19_A1     Q18_A1    Q17_A1
## Q45_A1 0.2761284 0.3769268 0.3022819 0.22269845 0.21491295 0.2141231
## Q35_A1 0.2649077 0.4926439 0.4359050 0.06522338 0.19917860 0.1857007
## Q36_A1 0.1791191 0.3814132 0.3645683 0.05165880 0.08809779 0.1222438
## Q44_A1 0.2442683 0.2477207 0.1139223 0.09263742 0.27117701 0.2642223
## Q42_A1 0.2148236 0.3440547 0.2750687 0.13514510 0.19934854 0.1676001
## Q43_A1 0.3682778 0.3910640 0.2814172 0.18961461 0.22752794 0.2517736
## Q21_A1 0.3147110 0.2289612 0.1694413 0.24115056 0.27266044 0.2378864
## Q22_A1 0.3325135 0.2424767 0.2308768 0.37631275 0.44803036 0.3806546
## Q30_A1 0.2874430 0.3430241 0.2580872 0.24771749 0.39922671 0.3667218
## Q31_A1 0.3373517 0.2808069 0.1694754 0.16010173 0.26248942 0.2481597
## Q40_A1 0.2426830 0.3496474 0.2476382 0.19068976 0.24135794 0.2320370
## Q41_A1 0.2868871 0.3349697 0.3129796 0.25517638 0.31122922 0.2689082
## Q39_A1 1.0000000 0.4636749 0.4907172 0.17236640 0.23228782 0.1771852
## Q37_A1 0.4636749 1.0000000 0.6153985 0.17768135 0.26305889 0.2202192
## Q38_A1 0.4907172 0.6153985 1.0000000 0.16735028 0.21003570 0.1647531
## Q19_A1 0.1723664 0.1776813 0.1673503 1.00000000 0.45174912 0.3333767
## Q18_A1 0.2322878 0.2630589 0.2100357 0.45174912 1.00000000 0.6368503
## Q17_A1 0.1771852 0.2202192 0.1647531 0.33337672 0.63685031 1.0000000
## Q20_A1 0.2256268 0.2395143 0.2133485 0.48790043 0.64443180 0.6600089
```

```
## Q33_A1 0.1532705 0.3219499 0.2641685 0.20068318 0.31374644 0.3206282
## Q32_A1 0.2934810 0.2342688 0.2342103 0.20876515 0.35141951 0.3804852
## Q26_A1 0.2904467 0.2992892 0.3140436 0.32672357 0.53534457 0.5217740
## Q27_A1 0.3612503 0.3295650 0.2915060 0.38272608 0.43756327 0.3933905
## Q25_A1 0.2278604 0.2749476 0.3079207 0.25241972 0.32046273 0.4006714
## Q28_A1 0.1788560 0.2372802 0.1920051 0.20021734 0.37553436 0.4187917
## Q29_A1 0.2139395 0.3218878 0.2645847 0.26173564 0.37443971 0.3808290
## Q34_A1 0.1665855 0.2709923 0.2479939 0.18092314 0.26943470 0.2398686
## Q23_A1 0.2538142 0.3741475 0.3479343 0.31381889 0.34706633 0.3341946
## Q24_A1 0.1759715 0.3854821 0.2596791 0.18149096 0.24570601 0.2840663
##            Q20_A1    Q33_A1    Q32_A1    Q26_A1    Q27_A1    Q25_A1
## Q45_A1 0.22974825 0.1981505 0.2219153 0.2727544 0.2994033 0.2712351
## Q35_A1 0.18740882 0.2287047 0.1755809 0.2107219 0.1869907 0.3247234
## Q36_A1 0.08543147 0.2378346 0.1287109 0.1372968 0.1321609 0.3116497
## Q44_A1 0.29168143 0.2481963 0.1904054 0.1889197 0.2500908 0.2519951
## Q42_A1 0.20125879 0.2748008 0.2059927 0.2481785 0.2295697 0.2105711
## Q43_A1 0.25599345 0.3378207 0.1862129 0.2727666 0.3372429 0.3080282
## Q21_A1 0.32625630 0.1497177 0.3151012 0.3695297 0.3780914 0.1327471
## Q22_A1 0.46007008 0.2000063 0.2992113 0.4973596 0.4431645 0.2948518
## Q30_A1 0.52962599 0.3838486 0.3827912 0.4887753 0.4739480 0.3525051
## Q31_A1 0.34197624 0.2327316 0.3902623 0.3534069 0.4161592 0.1632710
## Q40_A1 0.30462144 0.2110444 0.2386425 0.2627550 0.2678658 0.3339345
## Q41_A1 0.37325578 0.2230266 0.2883512 0.4009717 0.3420537 0.4282849
## Q39_A1 0.22562679 0.1532705 0.2934810 0.2904467 0.3612503 0.2278604
## Q37_A1 0.23951425 0.3219499 0.2342688 0.2992892 0.3295650 0.2749476
## Q38_A1 0.21334852 0.2641685 0.2342103 0.3140436 0.2915060 0.3079207
## Q19_A1 0.48790043 0.2006832 0.2087652 0.3267236 0.3827261 0.2524197
## Q18_A1 0.64443180 0.3137464 0.3514195 0.5353446 0.4375633 0.3204627
## Q17_A1 0.66000894 0.3206282 0.3804852 0.5217740 0.3933905 0.4006714
## Q20_A1 1.00000000 0.3298842 0.3910206 0.5823369 0.4753681 0.3918334
## Q33_A1 0.32988419 1.0000000 0.4193456 0.3836335 0.3964260 0.3919540
## Q32_A1 0.39102059 0.4193456 1.0000000 0.4935366 0.4623130 0.2349326
## Q26_A1 0.58233691 0.3836335 0.4935366 1.0000000 0.6000877 0.4288842
## Q27_A1 0.47536806 0.3964260 0.4623130 0.6000877 1.0000000 0.3172631
## Q25_A1 0.39183336 0.3919540 0.2349326 0.4288842 0.3172631 1.0000000
## Q28_A1 0.48610491 0.3239711 0.3194919 0.4889229 0.4379547 0.4147638
## Q29_A1 0.42238228 0.3137528 0.2951264 0.3690669 0.3970182 0.4287416
## Q34_A1 0.25453896 0.3429550 0.2810098 0.3005138 0.3046430 0.2657283
## Q23_A1 0.30817775 0.3522096 0.2758445 0.3627214 0.3984870 0.3663766
## Q24_A1 0.24316123 0.2742718 0.2282094 0.2727079 0.2944243 0.3303235
##           Q28_A1    Q29_A1    Q34_A1    Q23_A1    Q24_A1
## Q45_A1 0.2351469 0.3181362 0.2765228 0.3624822 0.2756742
## Q35_A1 0.1888195 0.1923297 0.1674686 0.3533225 0.3872615
## Q36_A1 0.1458032 0.2338024 0.1908820 0.2909794 0.2724822
## Q44_A1 0.2868591 0.3634032 0.1101782 0.2774968 0.2292726
## Q42_A1 0.2016138 0.2835501 0.2502479 0.2863765 0.2722631
## Q43_A1 0.3255304 0.4119503 0.2958574 0.3008900 0.2772823
## Q21_A1 0.2550409 0.1692463 0.1618518 0.2127192 0.1054845
## Q22_A1 0.2519636 0.2489104 0.1728229 0.2700329 0.1490491
## Q30_A1 0.4107304 0.3130594 0.2188967 0.2986267 0.2591695
## Q31_A1 0.2755025 0.2462785 0.2184258 0.1927703 0.2579554
## Q40_A1 0.1896739 0.2521386 0.1514853 0.3355144 0.3980677
## Q41_A1 0.3203282 0.3525574 0.2371236 0.3558968 0.2684365
## Q39_A1 0.1788560 0.2139395 0.1665855 0.2538142 0.1759715
```

```
## Q37_A1 0.2372802 0.3218878 0.2709923 0.3741475 0.3854821
## Q38_A1 0.1920051 0.2645847 0.2479939 0.3479343 0.2596791
## Q19_A1 0.2002173 0.2617356 0.1809231 0.3138189 0.1814910
## Q18_A1 0.3755344 0.3744397 0.2694347 0.3470663 0.2457060
## Q17_A1 0.4187917 0.3808290 0.2398686 0.3341946 0.2840663
## Q20_A1 0.4861049 0.4223823 0.2545390 0.3081778 0.2431612
## Q33_A1 0.3239711 0.3137528 0.3429550 0.3522096 0.2742718
## Q32_A1 0.3194919 0.2951264 0.2810098 0.2758445 0.2282094
## Q26_A1 0.4889229 0.3690669 0.3005138 0.3627214 0.2727079
## Q27_A1 0.4379547 0.3970182 0.3046430 0.3984870 0.2944243
## Q25_A1 0.4147638 0.4287416 0.2657283 0.3663766 0.3303235
## Q28_A1 1.0000000 0.6207771 0.2408956 0.3808010 0.3184995
## Q29_A1 0.6207771 1.0000000 0.4396733 0.4667551 0.3759791
## Q34_A1 0.2408956 0.4396733 1.0000000 0.4405226 0.3779485
## Q23_A1 0.3808010 0.4667551 0.4405226 1.0000000 0.5865368
## Q24_A1 0.3184995 0.3759791 0.3779485 0.5865368 1.0000000
```

Calcualte alphas, summary scales

Calculate Cronbach's alpha for English ability items (12-15) and study abroad stress (17-45)

```
psych::alpha(data[,str_c('Q', seq(12, 15), '_A1')])
```

```
##
## Reliability analysis
## Call: psych::alpha(x = data[, str_c("Q", seq(12, 15), "_A1")])
##
##   raw_alpha std.alpha G6(smc) average_r S/N   ase mean   sd
##       0.85      0.86    0.83       0.6   6 0.013  3.5 0.82
##
##  lower alpha upper     95% confidence boundaries
## 0.82 0.85 0.87
##
##  Reliability if an item is dropped:
##        raw_alpha std.alpha G6(smc) average_r S/N alpha se
## Q12_A1      0.79      0.80    0.73      0.56 3.9    0.019
## Q13_A1      0.80      0.80    0.74      0.58 4.1    0.019
## Q14_A1      0.86      0.86    0.81      0.68 6.4    0.013
## Q15_A1      0.79      0.81    0.76      0.58 4.1    0.020
##
##  Item statistics
##          n raw.r std.r r.cor r.drop mean   sd
## Q12_A1 361  0.86  0.87  0.83   0.74  3.5 0.92
## Q13_A1 361  0.84  0.86  0.81   0.73  3.5 0.86
## Q14_A1 361  0.79  0.76  0.63   0.59  3.3 1.13
## Q15_A1 361  0.86  0.86  0.79   0.73  3.6 1.00
##
## Non missing response frequency for each item
##           1    2    3    4    5 miss
## Q12_A1 0.02 0.10 0.41 0.32 0.14    0
## Q13_A1 0.02 0.07 0.43 0.35 0.12    0
## Q14_A1 0.06 0.16 0.40 0.20 0.19    0
## Q15_A1 0.03 0.10 0.32 0.37 0.18    0
```

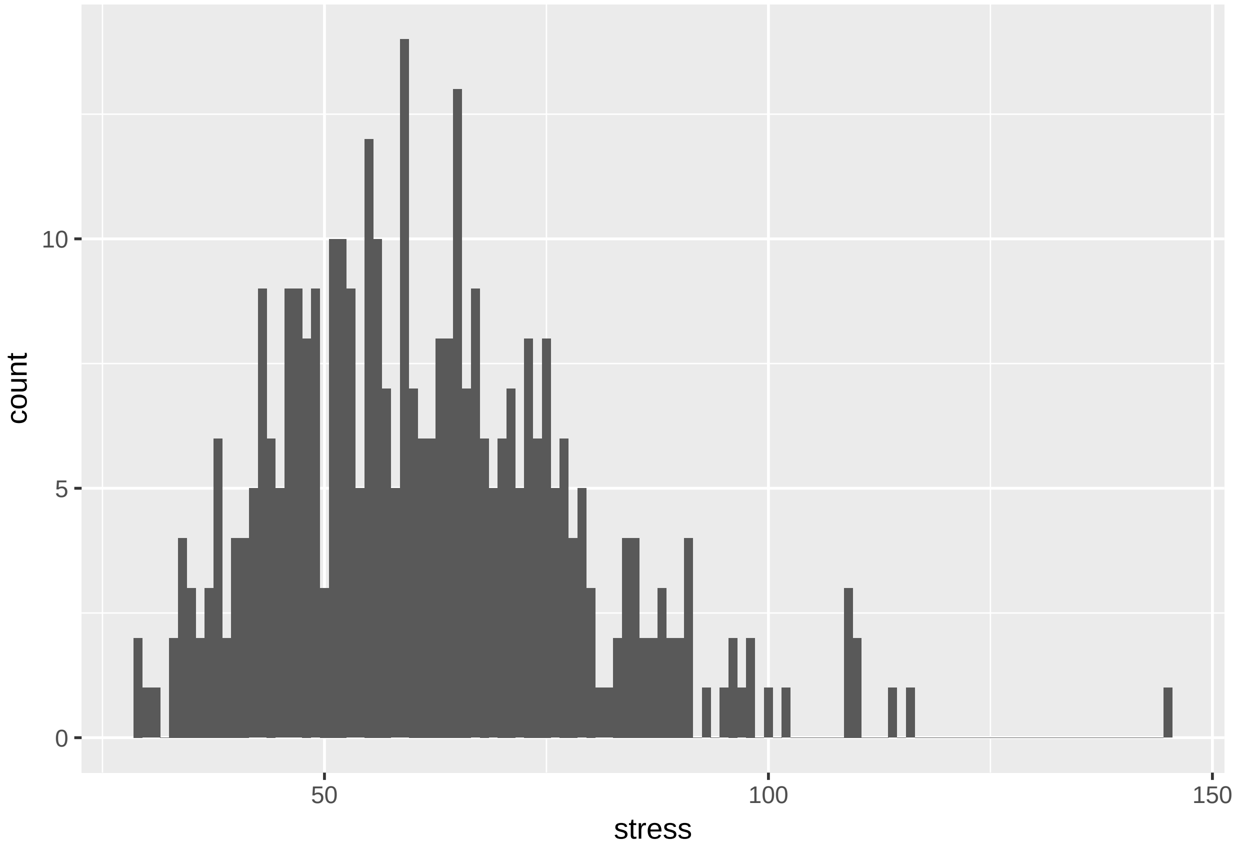
```r
psych::alpha(data[,str_c('Q', seq(17, 45), '_A1')])
```

```
##
## Reliability analysis
## Call: psych::alpha(x = data[, str_c("Q", seq(17, 45), "_A1")])
##
##   raw_alpha std.alpha G6(smc) average_r S/N    ase mean   sd
##       0.92      0.92    0.94      0.29  12 0.0061  2.1 0.59
##
##  lower alpha upper     95% confidence boundaries
## 0.91 0.92 0.93
##
##  Reliability if an item is dropped:
##        raw_alpha std.alpha G6(smc) average_r S/N alpha se
## Q17_A1      0.92      0.92    0.94      0.29  11   0.0063
## Q18_A1      0.92      0.92    0.94      0.29  11   0.0064
## Q19_A1      0.92      0.92    0.94      0.30  12   0.0062
## Q20_A1      0.91      0.92    0.94      0.29  11   0.0064
## Q21_A1      0.92      0.92    0.94      0.30  12   0.0062
## Q22_A1      0.92      0.92    0.94      0.29  12   0.0063
## Q23_A1      0.91      0.92    0.94      0.29  11   0.0064
## Q24_A1      0.92      0.92    0.94      0.29  12   0.0063
## Q25_A1      0.92      0.92    0.94      0.29  11   0.0064
## Q26_A1      0.91      0.92    0.94      0.29  11   0.0064
## Q27_A1      0.91      0.92    0.94      0.29  11   0.0064
## Q28_A1      0.92      0.92    0.94      0.29  11   0.0063
## Q29_A1      0.91      0.92    0.94      0.29  11   0.0064
## Q30_A1      0.92      0.92    0.94      0.29  11   0.0063
## Q31_A1      0.92      0.92    0.94      0.29  12   0.0062
## Q32_A1      0.92      0.92    0.94      0.29  12   0.0063
## Q33_A1      0.92      0.92    0.94      0.29  12   0.0063
## Q34_A1      0.92      0.92    0.94      0.29  12   0.0062
## Q35_A1      0.92      0.92    0.94      0.29  12   0.0063
## Q36_A1      0.92      0.92    0.94      0.30  12   0.0061
## Q37_A1      0.92      0.92    0.94      0.29  11   0.0064
## Q38_A1      0.92      0.92    0.94      0.29  12   0.0063
## Q39_A1      0.92      0.92    0.94      0.29  12   0.0062
## Q40_A1      0.92      0.92    0.94      0.29  12   0.0063
## Q41_A1      0.92      0.92    0.94      0.29  11   0.0064
## Q42_A1      0.92      0.92    0.94      0.29  12   0.0062
## Q43_A1      0.92      0.92    0.94      0.29  11   0.0063
## Q44_A1      0.92      0.92    0.94      0.30  12   0.0062
## Q45_A1      0.92      0.92    0.94      0.29  12   0.0062
##
##  Item statistics
##          n raw.r std.r r.cor r.drop mean   sd
## Q17_A1 361  0.59  0.60  0.59   0.55  2.4 1.05
## Q18_A1 361  0.61  0.62  0.61   0.57  2.2 1.07
## Q19_A1 361  0.46  0.46  0.44   0.40  2.6 1.16
## Q20_A1 361  0.65  0.67  0.67   0.62  1.8 0.92
## Q21_A1 361  0.41  0.45  0.43   0.37  1.3 0.73
## Q22_A1 361  0.52  0.55  0.53   0.48  1.5 0.87
## Q23_A1 361  0.66  0.64  0.63   0.62  2.7 1.15
## Q24_A1 361  0.57  0.55  0.53   0.52  2.3 1.24
```

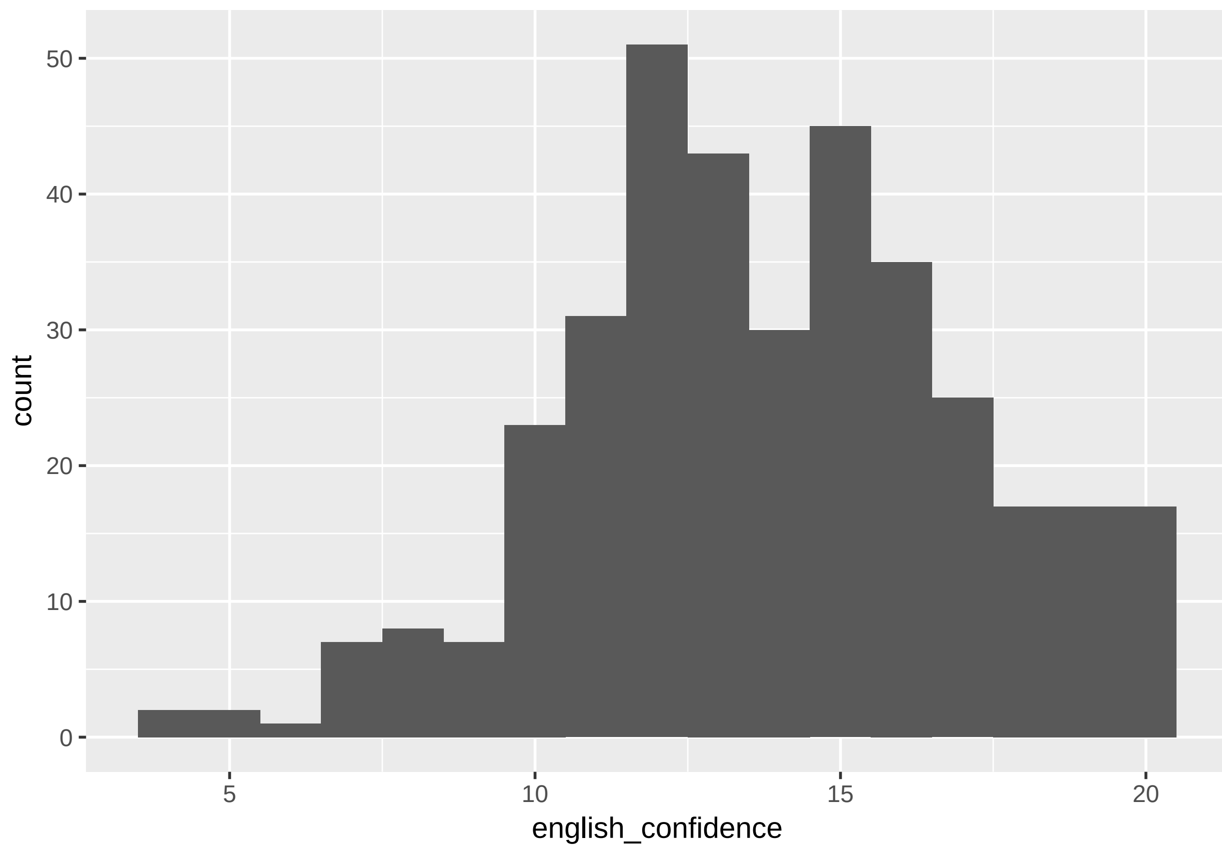
```
## Q25_A1 361  0.61  0.60  0.58   0.57  2.4 1.27
## Q26_A1 361  0.67  0.69  0.69   0.64  1.8 0.93
## Q27_A1 361  0.65  0.68  0.67   0.62  1.6 0.87
## Q28_A1 361  0.60  0.60  0.58   0.55  1.9 1.16
## Q29_A1 361  0.65  0.63  0.63   0.61  2.4 1.32
## Q30_A1 361  0.60  0.63  0.62   0.57  1.4 0.73
## Q31_A1 361  0.46  0.51  0.49   0.44  1.1 0.49
## Q32_A1 361  0.54  0.56  0.54   0.50  1.5 0.90
## Q33_A1 361  0.56  0.55  0.53   0.51  2.2 1.14
## Q34_A1 361  0.51  0.50  0.47   0.45  2.7 1.31
## Q35_A1 361  0.54  0.52  0.51   0.49  3.0 1.14
## Q36_A1 361  0.45  0.42  0.41   0.40  3.4 1.18
## Q37_A1 361  0.62  0.62  0.61   0.58  1.9 1.00
## Q38_A1 361  0.54  0.54  0.52   0.49  2.3 1.07
## Q39_A1 361  0.50  0.52  0.50   0.46  1.5 0.88
## Q40_A1 361  0.54  0.52  0.50   0.48  2.3 1.22
## Q41_A1 361  0.61  0.60  0.59   0.57  1.9 1.11
## Q42_A1 361  0.51  0.49  0.46   0.45  2.6 1.39
## Q43_A1 361  0.60  0.60  0.58   0.56  1.8 1.00
## Q44_A1 361  0.47  0.47  0.44   0.42  2.3 1.10
## Q45_A1 361  0.53  0.51  0.48   0.47  2.8 1.20
##
## Non missing response frequency for each item
##           1    2    3    4    5 miss
## Q17_A1 0.25 0.28 0.33 0.11 0.02    0
## Q18_A1 0.30 0.34 0.22 0.11 0.02    0
## Q19_A1 0.22 0.22 0.33 0.18 0.05    0
## Q20_A1 0.44 0.34 0.18 0.03 0.02    0
## Q21_A1 0.78 0.15 0.04 0.02 0.01    0
## Q22_A1 0.68 0.18 0.09 0.04 0.01    0
## Q23_A1 0.19 0.25 0.33 0.16 0.07    0
## Q24_A1 0.36 0.21 0.26 0.11 0.06    0
## Q25_A1 0.34 0.21 0.24 0.14 0.07    0
## Q26_A1 0.47 0.30 0.18 0.04 0.01    0
## Q27_A1 0.62 0.21 0.14 0.02 0.01    0
## Q28_A1 0.51 0.21 0.16 0.09 0.04    0
## Q29_A1 0.38 0.18 0.20 0.17 0.07    0
## Q30_A1 0.70 0.22 0.06 0.02 0.00    0
## Q31_A1 0.92 0.04 0.02 0.00 0.01    0
## Q32_A1 0.69 0.15 0.11 0.04 0.01    0
## Q33_A1 0.35 0.25 0.26 0.10 0.04    0
## Q34_A1 0.26 0.19 0.28 0.16 0.11    0
## Q35_A1 0.11 0.19 0.40 0.18 0.12    0
## Q36_A1 0.07 0.14 0.32 0.25 0.22    0
## Q37_A1 0.45 0.30 0.19 0.04 0.02    0
## Q38_A1 0.29 0.32 0.26 0.10 0.03    0
## Q39_A1 0.67 0.18 0.12 0.02 0.01    0
## Q40_A1 0.37 0.20 0.27 0.11 0.05    0
## Q41_A1 0.48 0.24 0.19 0.06 0.04    0
## Q42_A1 0.30 0.19 0.26 0.10 0.15    0
## Q43_A1 0.47 0.32 0.15 0.04 0.03    0
## Q44_A1 0.29 0.28 0.31 0.08 0.04    0
## Q45_A1 0.17 0.21 0.34 0.17 0.10    0
```

Alpha's are good! Let's calculate scales (english confidence and stress levels) and plot the dependent measure (stress)

```
data <- data %>% mutate(english_confidence = rowSums(.[str_c('Q', seq(12, 15), '_A1')]))
data <- data %>% mutate(stress = rowSums(.[str_c('Q', seq(17, 45), '_A1')]))

print(ggplot(data, aes(stress)) + geom_histogram(binwidth = 1))
```




```
print(ggplot(data, aes(english_confidence)) + geom_histogram(binwidth = 1)) + labs(x="English confidenc
```



count
english_confidence
0
10
20
30
40
50
5
10
15
20

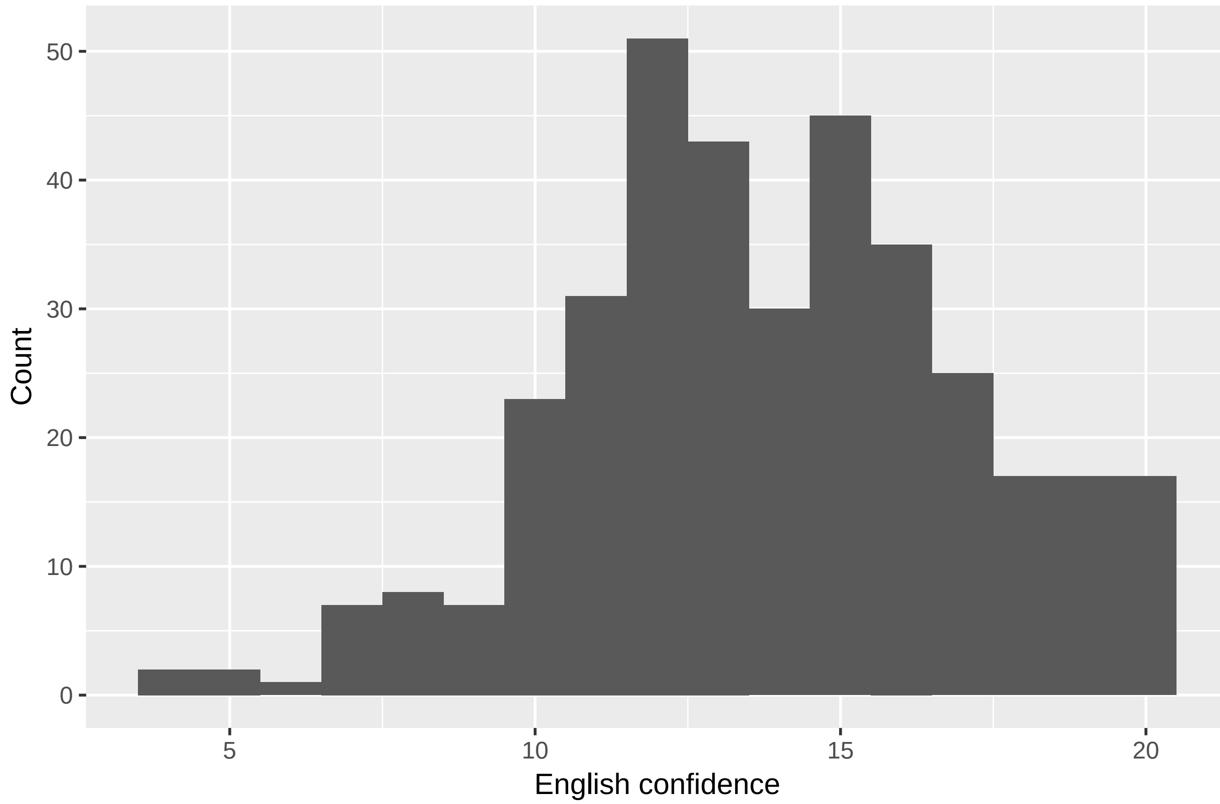


Figure 4: Histogram of English confidence scores

Looks like an outlier at ~ 140, let's filter and re-plot. Then let's see what the highest correlations with stress are.

```
filtered <- data %>% filter(stress < 130)
ggplot(filtered, aes(stress)) + geom_histogram() + labs(x="Stress", y="Count", caption = "Figure 5: His
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

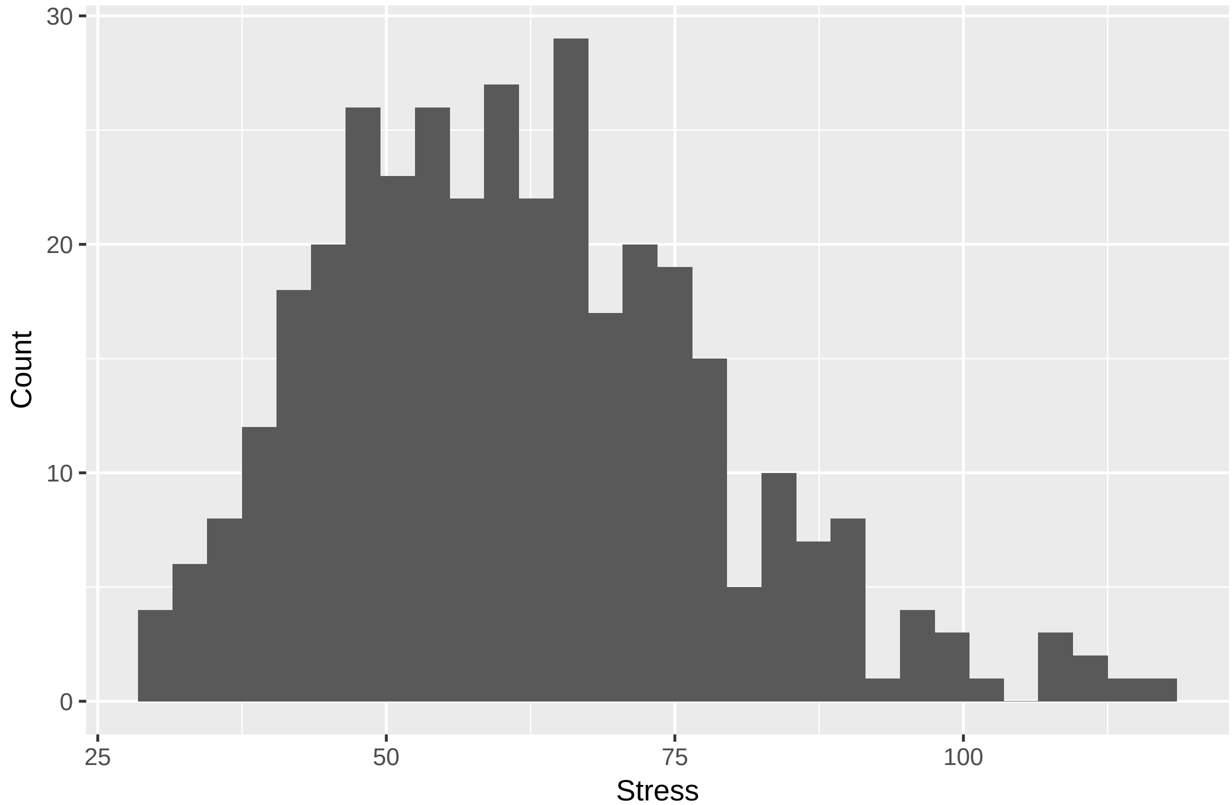


Figure 5: Histogram of stress scores

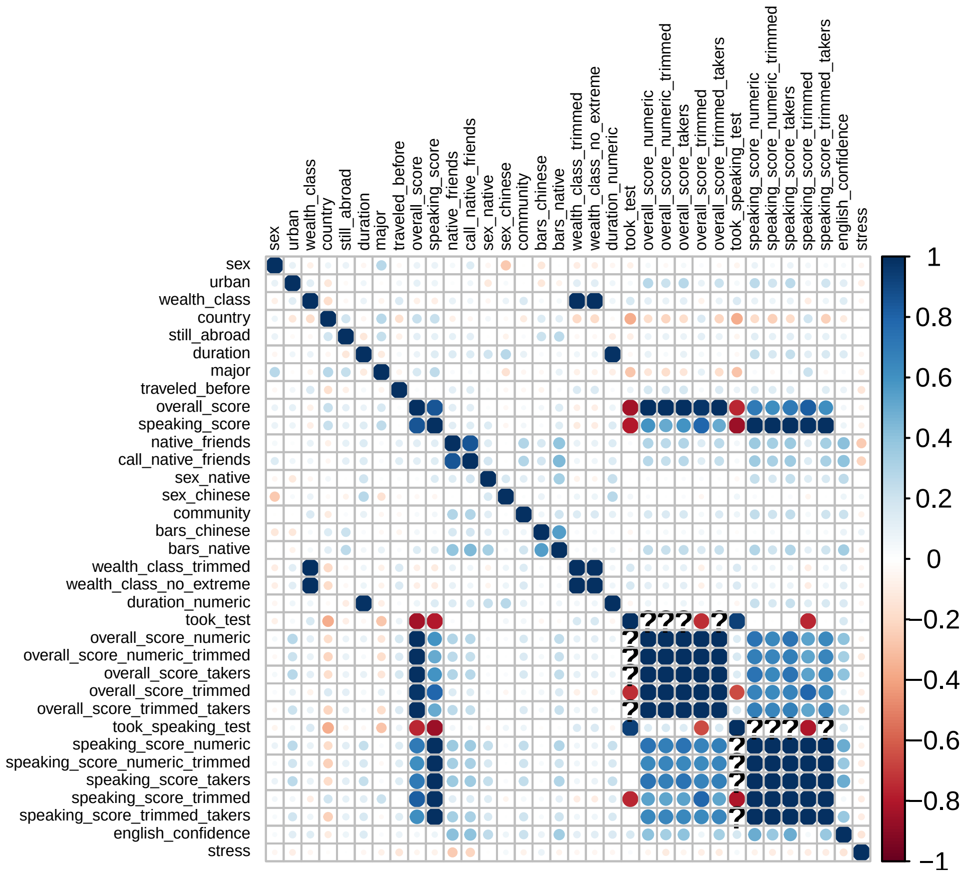
Looks reasonably normally-distributed, might benefit from a log transformation but… We should probably re-calculate alpha's as well, with the filtered data.

But first let's replot our correlations plot to look for patterns in the data with our new summary measures.

```
elim <- c(str_c('Q',seq(12, 15), '_A1'), str_c('Q', seq(17, 49), '_A1'))
numeric_vars <- dplyr::select(filtered, -c(1:7)) %>% dplyr::select(which(!colnames(.) %in% elim)) %>% mu
correls <- cor(numeric_vars,  use = "pairwise.complete.obs")
```

```
## Warning in cor(numeric_vars, use = "pairwise.complete.obs"): the standard
## deviation is zero
```

```
corrplot(correls,  tl.cex = .5 , tl.col = "black")
```




```r
sort(abs(correls[,'stress']))
```

```
##                still_abroad                  sex_native
##                4.846593e-05                9.590382e-03
##                     country                   took_test
##                1.775013e-02                2.741645e-02
##          took_speaking_test            duration_numeric
##                4.161986e-02                4.260119e-02
##        wealth_class_trimmed                    duration
##                4.494875e-02                4.599403e-02
##     wealth_class_no_extreme                wealth_class
##                4.614176e-02                4.630526e-02
##                 sex_chinese                         sex
##                4.671088e-02                4.962594e-02
##               overall_score       overall_score_numeric
##                5.328315e-02                5.878870e-02
##        overall_score_takers                 bars_native
##                5.878870e-02                6.066555e-02
##                   community      speaking_score_numeric
##                6.768621e-02                6.890142e-02
##       speaking_score_takers              speaking_score
##                6.890142e-02                6.897921e-02
##                       major                bars_chinese
##                7.532526e-02                8.370950e-02
##       overall_score_trimmed                       urban
##                8.617199e-02                8.685166e-02
```

```
##         speaking_score_trimmed speaking_score_numeric_trimmed
##                   9.343179e-02                   9.507400e-02
##  speaking_score_trimmed_takers  overall_score_numeric_trimmed
##                   9.507400e-02                   1.044113e-01
##   overall_score_trimmed_takers                traveled_before
##                   1.044113e-01                   1.315451e-01
##             english_confidence             call_native_friends
##                   1.343782e-01                   2.143226e-01
##                 native_friends                         stress
##                   2.571252e-01                   1.000000e+00
```

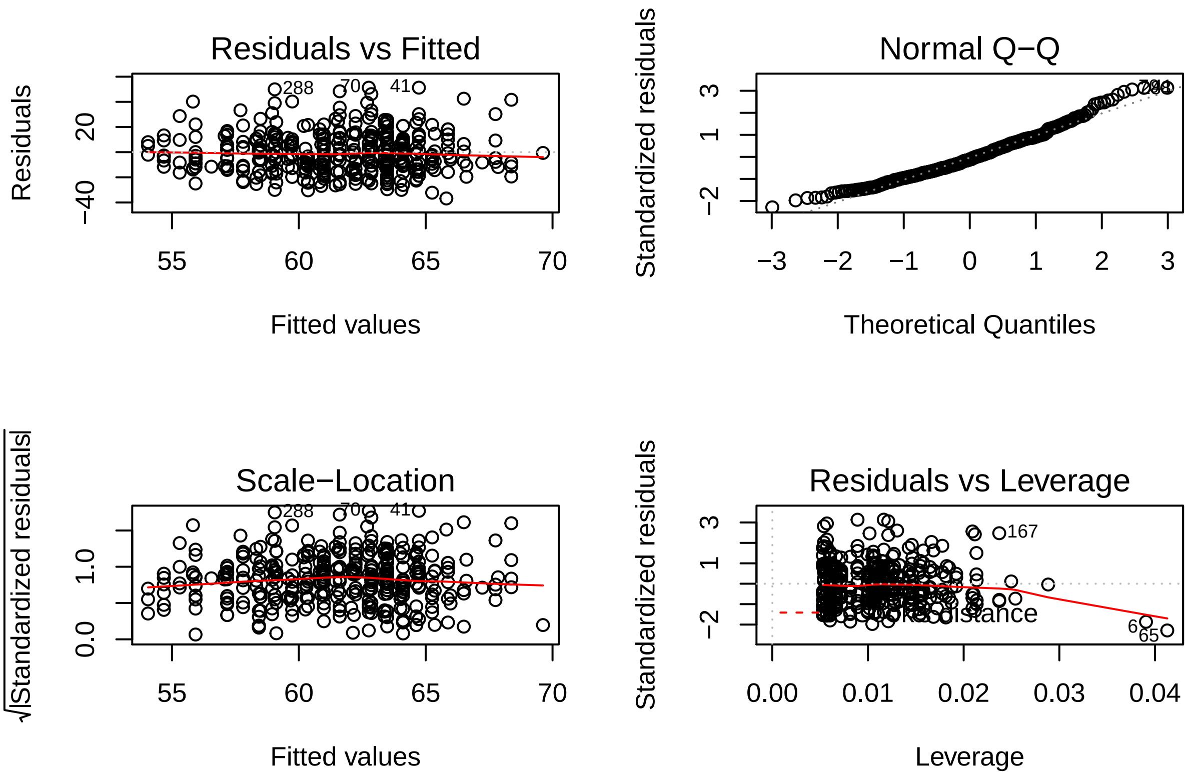
Simple linear models involving English confidence

. . . we'll look at the residuals from the statistical models before attempting that. Try a simple linear model first, predicting reported stress levels from our english confidence measure, including sex and whether the subject has traveled before as covariates.

```r
stressModel <- lm(stress ~ english_confidence + sex + traveled_before, data = filtered)
summary(stressModel)
```

```
##
## Call:
## lm(formula = stress ~ english_confidence + sex + traveled_before,
##     data = filtered)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -36.816 -11.599  -1.843  10.526  51.267
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   72.7467     3.9543  18.397   <2e-16 ***
## english_confidence            -0.6243     0.2679  -2.330   0.0204 *
## sexFemale                     -1.7706     1.8477  -0.958   0.3386
## traveled_beforeAbroad before  -4.4336     1.8889  -2.347   0.0195 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.43 on 356 degrees of freedom
## Multiple R-squared:  0.03501,    Adjusted R-squared:  0.02688
## F-statistic: 4.305 on 3 and 356 DF,  p-value: 0.005322
```

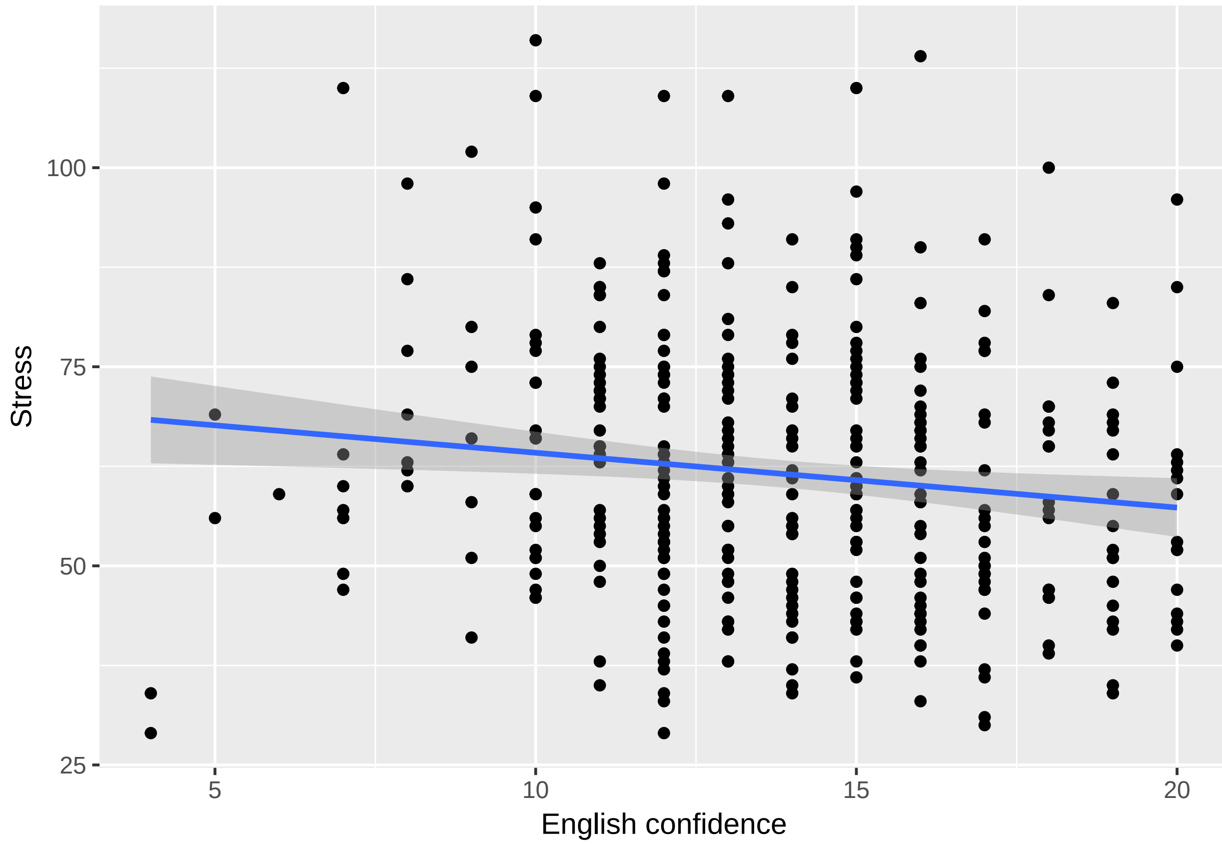
```r
par(mfrow=c(2,2))
plot(stressModel)
```



Looks like english confidence and whether a person has traveled before are both significant predictors, with the former in the negative direction (greater confidence associated less stress) and with those who have not traveled before experiencing more stress (as expected).

A nicer plot of the linear model:

```
ggplot(filtered, aes(x=english_confidence, y=stress)) + geom_point() + geom_smooth(method="lm") + labs(
```



Two points on the bottom left look like multivariate outliers, probably have high leverage on the regression equation. Let's refit the model without these 2 and re-plot.

```
refiltered <- filtered %>% filter(english_confidence > 4)
stressModel <- lm(stress ~ english_confidence + sex + traveled_before, data = refiltered)
summary(stressModel)
```

```
##
## Call:
## lm(formula = stress ~ english_confidence + sex + traveled_before,
##     data = refiltered)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -32.656 -11.806  -1.836  10.589  51.624
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                     75.4947     4.0182  18.788  < 2e-16 ***
## english_confidence              -0.8199     0.2731  -3.002  0.00287 **
## sexFemale                       -1.8944     1.8353  -1.032  0.30268
## traveled_beforeAbroad before   -3.6614     1.8875  -1.940  0.05320 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.27 on 354 degrees of freedom
## Multiple R-squared:  0.04185,    Adjusted R-squared:  0.03373
```

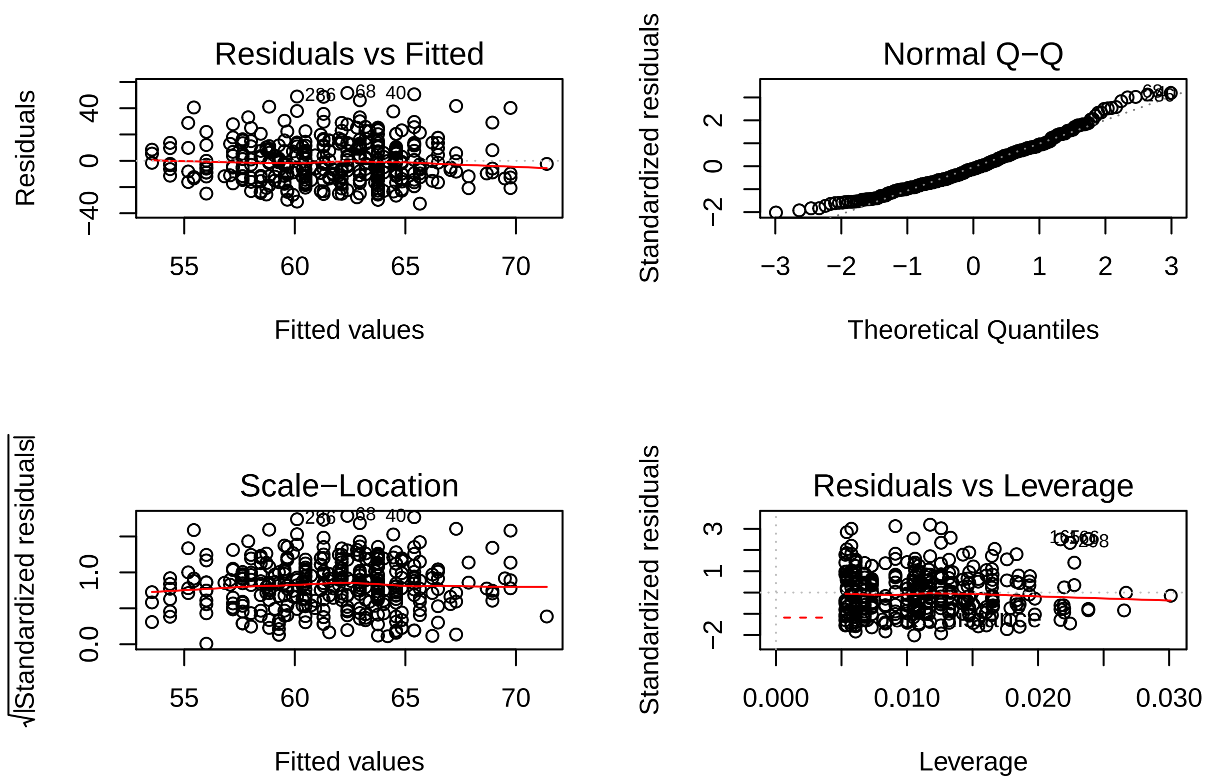
	Stress
(Constant)	**75.495*****
	(4.018)
English confidence	$-$**0.820****
	(0.273)
Gender (Female)	-1.894
	(1.835)
Prior travel	-3.661
	(1.888)
R^2	0.042
Adj. R^2	0.034
Num. obs.	358

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$.
Results are as coefficient (standard error).

Table 2: English confidence predicts stress scores

```
## F-statistic: 5.154 on 3 and 354 DF,  p-value: 0.001688
```

```r
par(mfrow=c(2,2))
plot(stressModel)
```



Residual plots look pretty good now (not perfect - still might try log transformation). Let's make a nice looking tabular output of the statistical results.

```r
texreg(stressModel, type = "html",  digits = 3, bold = .05, booktabs = TRUE, sideways = FALSE, use.packa
```

```r
ggplot(refiltered, aes(x=english_confidence, y=stress)) + geom_point() + geom_smooth(method="lm") + labs
```

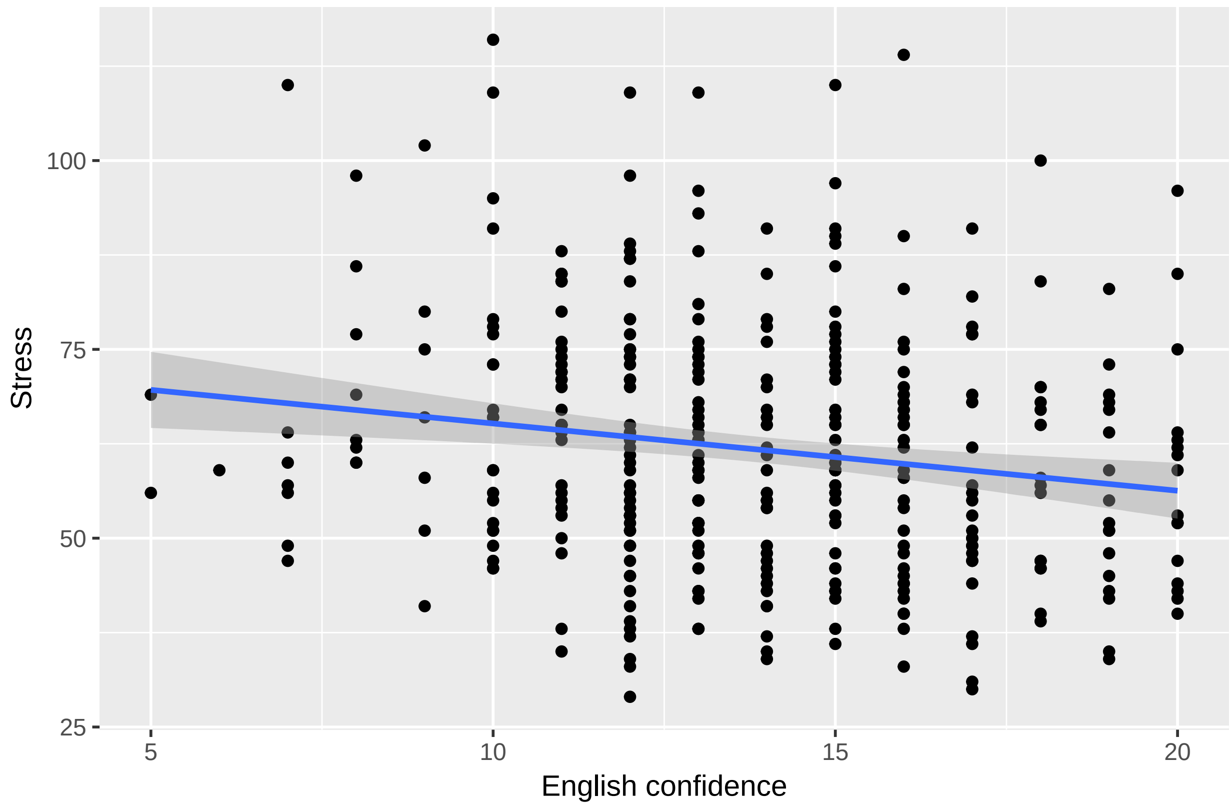


Figure 6: Linear relationship between English confidence and stress

So it seems there's a nice linear relationship between English confidence and reported levels. The regression output tells us that for a 1 unit increase in "English confidence level" we can expect about a 0.82 point decrease in reported stress level. Whether a participant has traveled before is no longer a significant predictor, but it is very close (sex remains non-significant).

Let's also make a correlation matrix between potential covariates and stress.

```r
numeric_vars_filtered <- refiltered %>% dplyr::select(stress, still_abroad, duration_numeric, wealth_cla
numeric_vars_filtered$traveled_before <- numeric_vars_filtered$traveled_before - 1
numeric_vars_filtered$still_abroad <- numeric_vars_filtered$still_abroad - 1
numeric_vars_filtered$took_test <- numeric_vars_filtered$took_test - 1
numeric_vars_filtered$sex <- numeric_vars_filtered$sex - 1
numeric_vars_filtered$urban <- numeric_vars_filtered$urban - 1
chart.Correlation(R = numeric_vars_filtered , histogram = FALSE, pch = 19)
```

```
## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter
```

```
## Warning in axis(side = side, at = at, labels = labels, ...): "method" is
## not a graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "method" is
## not a graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "method" is
## not a graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "method" is
## not a graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "method" is
## not a graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "method" is
## not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
```

```
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
```

```
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in axis(side = side, at = at, labels = labels, ...): "method" is
## not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in axis(side = side, at = at, labels = labels, ...): "method" is
## not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
```

```
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
```

```
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in axis(side = side, at = at, labels = labels, ...): "method" is
## not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in axis(side = side, at = at, labels = labels, ...): "method" is
## not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
```

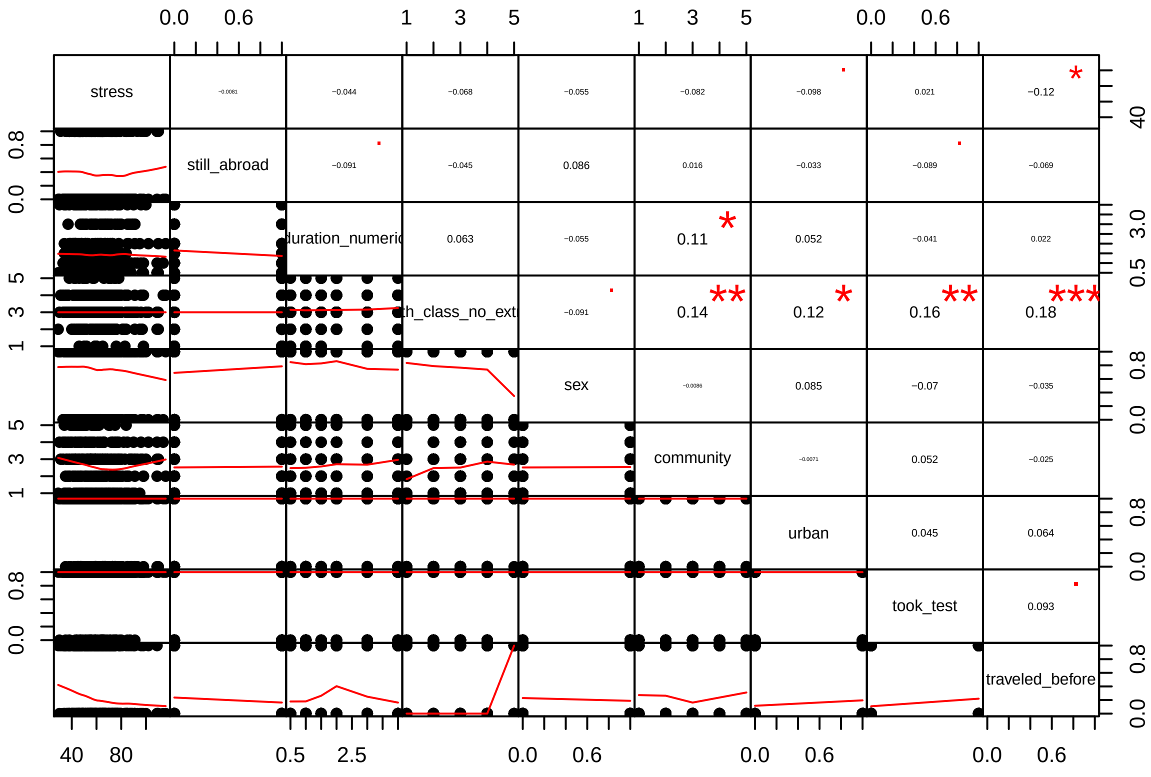
```
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
```

```
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in axis(side = side, at = at, labels = labels, ...): "method" is
## not a graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in axis(side = side, at = at, labels = labels, ...): "method" is
## not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter
## Warning in plot.window(...): "method" is not a graphical parameter
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter
## Warning in title(...): "method" is not a graphical parameter
```

```
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "method" is
## not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter
```

```
## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "method" is
## not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "method" is
## not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "method" is
## not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter

## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in plot.xy(xy.coords(x, y), type = type, ...): "method" is not a
## graphical parameter

## Warning in plot.window(...): "method" is not a graphical parameter
```

```
## Warning in plot.xy(xy, type, ...): "method" is not a graphical parameter

## Warning in title(...): "method" is not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "method" is
## not a graphical parameter

## Warning in axis(side = side, at = at, labels = labels, ...): "method" is
## not a graphical parameter
```



And we'll, just to be safe, do a regression with all these control variables to show that the regression parameters don't change much when they're included.

```
eng_conf_with_covs <- lm(stress ~ english_confidence + sex + traveled_before + still_abroad + duration_
summary(eng_conf_with_covs)
```

```
##
## Call:
## lm(formula = stress ~ english_confidence + sex + traveled_before +
##     still_abroad + duration_numeric + wealth_class_no_extreme +
##     urban, data = refiltered)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -33.326 -11.946  -1.448   9.910  51.409
##
## Coefficients:
##                                     Estimate Std. Error t value
```

```
## (Intercept)                              77.8792     6.1287  12.707
## english_confidence                       -0.7583     0.2813  -2.696
## sexFemale                                -1.8655     1.8786  -0.993
## traveled_beforeAbroad before             -3.3587     1.9424  -1.729
## still_abroadAlready returned             -0.2415     1.8055  -0.134
## duration_numeric                         -0.1722     0.8188  -0.210
## wealth_class_no_extremeLower-middle class  1.6333     5.3138   0.307
## wealth_class_no_extremeMiddle class      -1.3658     4.9453  -0.276
## wealth_class_no_extremeUpper-middle class  -0.2069     5.0900  -0.041
## wealth_class_no_extremeUpper class       -3.1692     6.4791  -0.489
## urbanUrban                               -2.7771     2.3345  -1.190
##                                          Pr(>|t|)
## (Intercept)                              < 2e-16 ***
## english_confidence                       0.00736 **
## sexFemale                                0.32140
## traveled_beforeAbroad before             0.08468 .
## still_abroadAlready returned             0.89367
## duration_numeric                         0.83358
## wealth_class_no_extremeLower-middle class  0.75875
## wealth_class_no_extremeMiddle class      0.78258
## wealth_class_no_extremeUpper-middle class  0.96760
## wealth_class_no_extremeUpper class       0.62505
## urbanUrban                               0.23502
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.35 on 347 degrees of freedom
## Multiple R-squared:  0.05176,    Adjusted R-squared:  0.02443
## F-statistic: 1.894 on 10 and 347 DF,  p-value: 0.04492
```

Now a regression output in a LaTeX table for this model.

```
#texreg(eng_conf_with_covs, type = "html",  digits = 3, bold = .05, booktabs = TRUE, sideways = FALSE,
```

Objective measures; IELTS / TOEFL scores, country effects

Next we're interested in what other factors we've collected data on are predictors of stress - ideally, factors that are more 'objective' measures, such as IELTS/TOEFL scores.

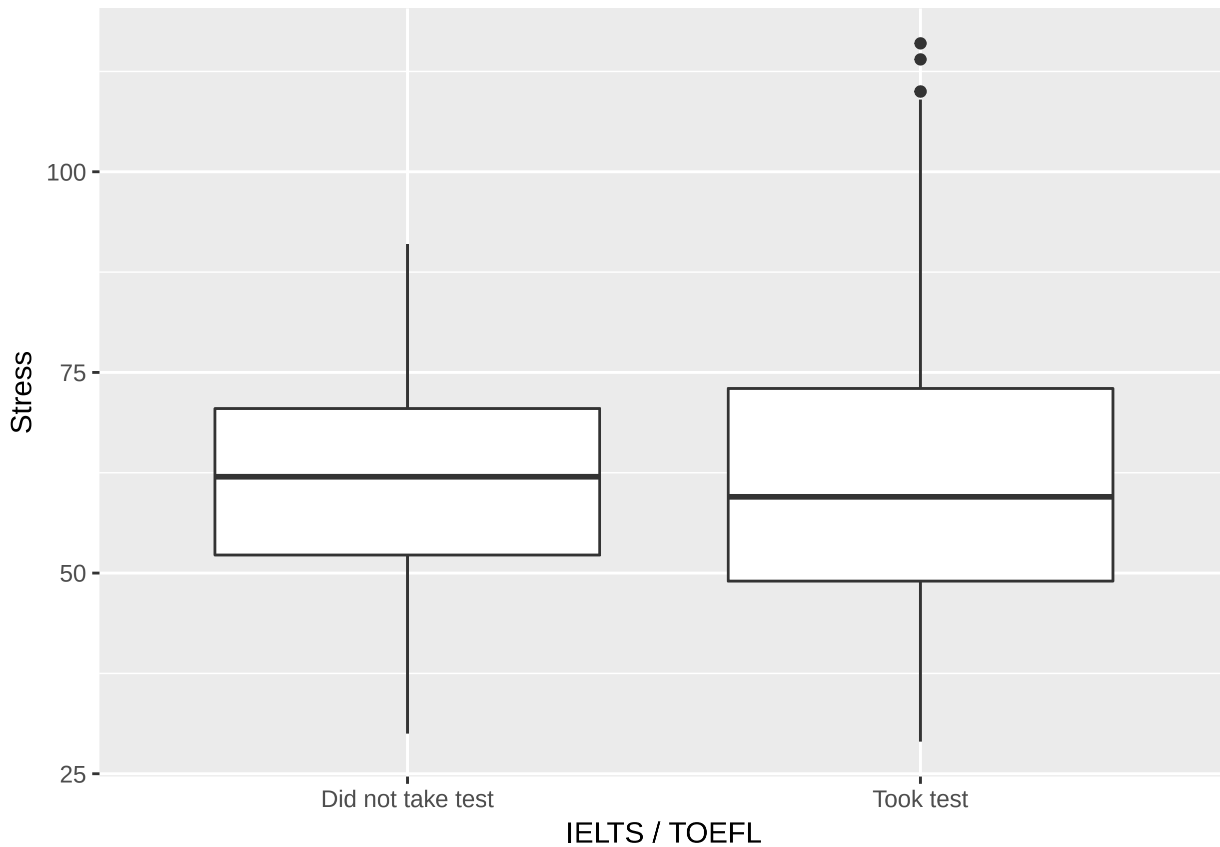
To start off we'll look at whether those who took the test experienced less stress studying abroad than those that did. And we'll take a look at the counts for number of subjects who did and did not take the test.

```
took_test_model <- lm(stress ~ took_test + sex + traveled_before, data  = refiltered)
summary(took_test_model)
```

```
##
## Call:
## lm(formula = stress ~ took_test + sex + traveled_before, data = refiltered)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -31.660 -11.949  -1.844  10.672  53.357
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
```

```
## (Intercept)                     63.466      2.493  25.462   <2e-16 ***
## took_testTook test               1.195      2.259   0.529   0.5973
## sexFemale                       -2.018      1.861  -1.084   0.2790
## traveled_beforeAbroad before    -4.420      1.905  -2.320   0.0209 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.47 on 354 degrees of freedom
## Multiple R-squared:  0.01823,    Adjusted R-squared:  0.009909
## F-statistic: 2.191 on 3 and 354 DF,  p-value: 0.08881
```

```r
ggplot(refiltered, aes(x=took_test, y=stress)) + geom_boxplot() + labs(x = "IELTS / TOEFL", y="Stress")
```




```r
table(refiltered$took_test)
```

```
##
## Did not take test         Took test
##                66               292
```

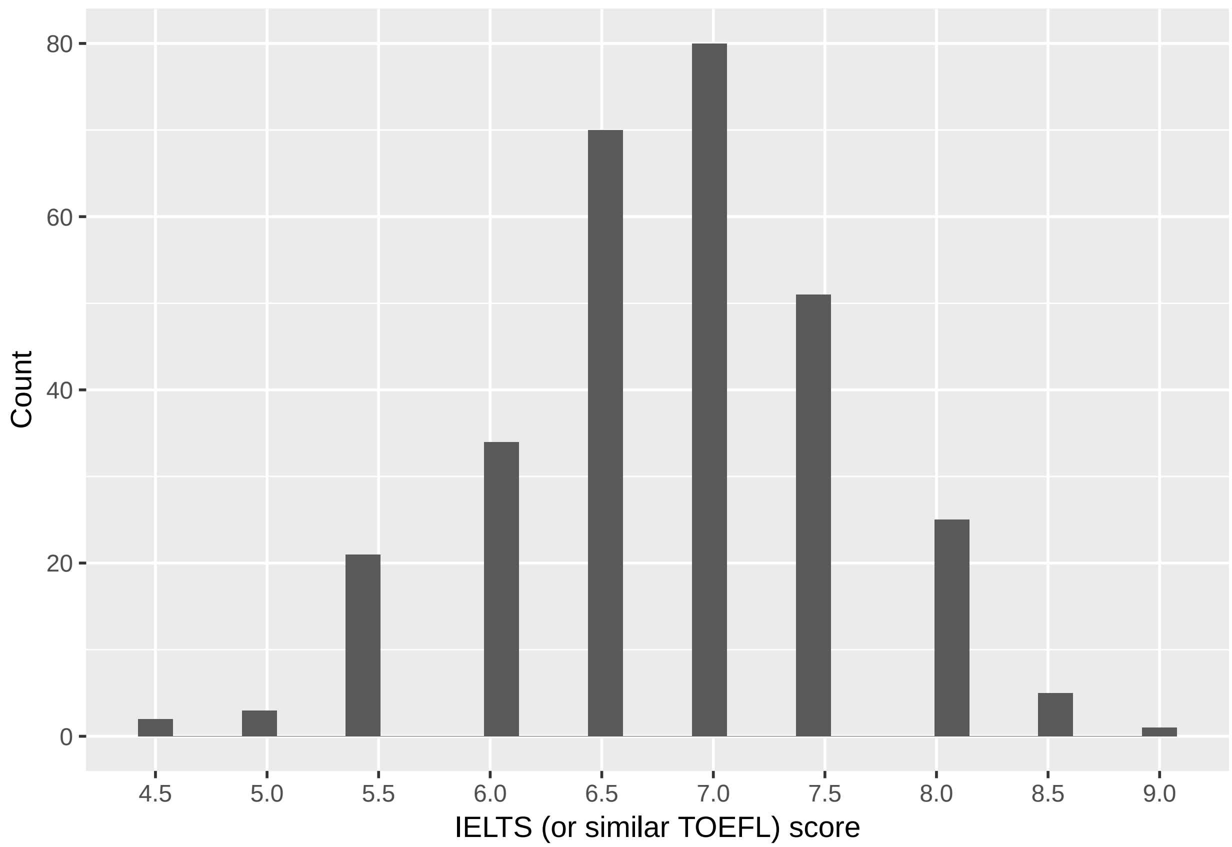
Looks like stress levels didn't differ much according to whether one took the test or not. This doesn't tell us much about how doing well on the test relates to stress, so next, let's look at how they relate to the scores themselves (amongst those that took the test). But first we'll generate a histogram to see our distribution of scores, because we suspect that there many not be many people who received the highest and lowest scores, so those data points may not be very informative in our statistical models. We'll also produce a table of the counts.

```r
table(refiltered$overall_score_numeric)
```

```
##
## 4.5   5 5.5   6 6.5   7 7.5   8 8.5   9
##   2   3  21  34  70  80  51  25   5   1
```

```
ggplot(refiltered, aes(x=overall_score_numeric)) + geom_histogram() + scale_x_continuous("IELTS (or sim
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.

## Warning: Removed 66 rows containing non-finite values (stat_bin).
```



As we suspected, very few subjects received the highest scores (> 8.0) or the lowest (< 5). Let's exclude them from our analyses since they're outliers and it's difficult for us to infer about them. This means we'll exclude 8 subjects, plus the 66 subjects who did not report taking the test. This is given in our variable `overall_score_numeric_less_trimmed` which we calculated in the first few code blocks. First let's analyze whether score predicts stress levels.

```
refiltered$overall_score_numeric_less_trimmed <- refiltered$overall_score_numeric
refiltered$overall_score_numeric_less_trimmed[refiltered$overall_score_numeric_less_trimmed == 9] <- NA
refiltered$overall_score_numeric_less_trimmed[refiltered$overall_score_numeric_less_trimmed == 4.5] <- N
score_model <- lm(stress ~ overall_score_numeric_less_trimmed + sex + traveled_before, data = refiltere
summary(score_model)
```

```
##
## Call:
## lm(formula = stress ~ overall_score_numeric_less_trimmed + sex +
##     traveled_before, data = refiltered)
##
## Residuals:
```

	Stress
(Constant)	**74.237*****
	(9.668)
Test scores	-1.528
	(1.413)
Gender (Female)	-0.957
	(2.124)
Prior travel	$-$**4.571***
	(2.197)
R^2	0.023
Adj. R^2	0.013
Num. obs.	289

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$.
Results are as coefficient (standard error).

Table 3: No overall relationship between test scores and stress

```
##     Min      1Q  Median      3Q     Max
## -31.307 -13.114  -2.072  10.413  52.650
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                         74.2371     9.6680   7.679 2.59e-13
## overall_score_numeric_less_trimmed  -1.5276     1.4132  -1.081   0.2806
## sexFemale                           -0.9573     2.1244  -0.451   0.6526
## traveled_beforeAbroad before        -4.5706     2.1973  -2.080   0.0384
##
## (Intercept)                        ***
## overall_score_numeric_less_trimmed
## sexFemale
## traveled_beforeAbroad before       *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.05 on 285 degrees of freedom
##   (69 observations deleted due to missingness)
## Multiple R-squared:  0.02346,    Adjusted R-squared:  0.01318
## F-statistic: 2.282 on 3 and 285 DF,  p-value: 0.07935
```

```r
texreg(score_model, type = "html",  digits = 3, bold = .05, booktabs = TRUE, sideways = FALSE, use.packa
```

So overall score isn't a significant predictor at a group level. We're a little surprised by this result, given that, as shown below, IELTS / TOEFL score is correlated with English confidence, which was associated with stress levels (we'll come back to this point):

```r
cor.test(refiltered$overall_score_numeric_less_trimmed, refiltered$english_confidence)
```

```
##
##  Pearson's product-moment correlation
##
## data:  refiltered$overall_score_numeric_less_trimmed and refiltered$english_confidence
## t = 6.6503, df = 287, p-value = 1.474e-10
## alternative hypothesis: true correlation is not equal to 0
## 95 percent confidence interval:
```

```
##  0.2610370 0.4613391
## sample estimates:
##       cor
## 0.3654104
```

So what might account for this? We hypothesized that the impact of test scores might differ according to where a subject studied abroad - the country where they stayed. First let's graph stress levels by country, and test whether there are differences by country.

```r
ggplot(refiltered, aes(x=country, y=stress)) + geom_boxplot() + labs(x="Country", y="Stress" , caption =
```

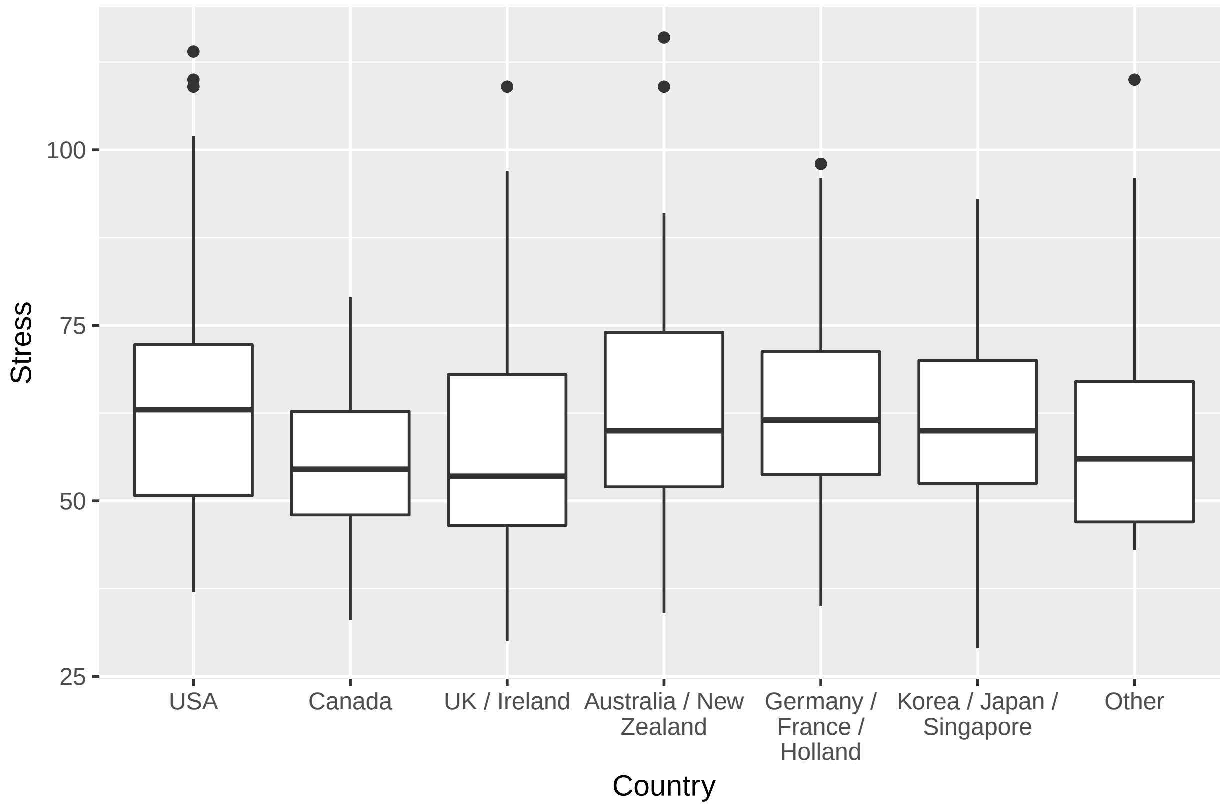


Figure 7: Stress levels reported by country

```r
stress_by_country <- lm(stress ~ country, data = refiltered)
summary(stress_by_country)
```

```
##
## Call:
## lm(formula = stress ~ country, data = refiltered)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -32.875 -11.702  -1.875   9.844  53.178
##
## Coefficients:
##                                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)                           64.094      1.686  38.007   <2e-16 ***
## countryCanada                         -7.821      3.906  -2.003   0.0460 *
## countryUK / Ireland                   -5.518      2.642  -2.089   0.0375 *
## countryAustralia / New Zealand        -1.272      2.566  -0.496   0.6204
```

	Stress
(Constant)	**64.094*****
	(1.686)
Canada - USA	$-$**7.821***
	(3.906)
]UK / Ireland - USA	$-$**5.518***
	(2.642)
Australia / New Zealand - USA	-1.272
	(2.566)
Germany / France / Holland - USA	-1.753
	(3.008)
Korea / Japan / Singapore - USA	-2.219
	(3.109)
Other - USA	-2.623
	(4.348)
R^2	0.020
Adj. R^2	0.003
Num. obs.	358

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$.
Results are as coefficient (standard error).

Table 4: Differences in stress scores from USA by country

```
## countryGermany / France / Holland   -1.753      3.008  -0.583   0.5605
## countryKorea / Japan / Singapore    -2.219      3.109  -0.714   0.4760
## countryOther                        -2.623      4.348  -0.603   0.5467
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.52 on 351 degrees of freedom
## Multiple R-squared:  0.01998,    Adjusted R-squared:  0.003227
## F-statistic: 1.193 on 6 and 351 DF,  p-value: 0.3095
```

```
Anova(stress_by_country, type = "III")
```

```
## Anova Table (Type III tests)
##
## Response: stress
##             Sum Sq  Df   F value Pr(>F)
## (Intercept) 394369   1 1444.5621 <2e-16 ***
## country       1954   6    1.1926 0.3095
## Residuals    95824 351
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

So it does look like there's a little heterogeneity in stress levels according to country, with USA students experiencing the most stress. There's not an overall effect of country but there are a few differences from the US.

```
texreg(stress_by_country, type = "html",  digits = 3, bold = .05, booktabs = TRUE, sideways = FALSE, us
```

We'll proceed with our analysis of whether test scores' relationship with stress levels differ by county (`overall_score_numeric_trimmed:country` interaction), excluding subjects who responded "Other" since subjects in this group don't necessarily form a meaningful / cohesive group for statistical analysis.

```r
noOtherCountry <- refiltered %>% filter(country != "Other")
countryModel <- lm(stress ~ overall_score_numeric_less_trimmed * country + sex + traveled_before, data =
Anova(countryModel, type = "III")
```

```
## Anova Table (Type III tests)
##
## Response: stress
##                                            Sum Sq  Df F value    Pr(>F)
## (Intercept)                                 10266   1 36.3856 5.469e-09
## overall_score_numeric_less_trimmed           1983   1  7.0301  0.008502
## country                                      2665   5  1.8891  0.096522
## sex                                            15   1  0.0549  0.814979
## traveled_before                              1084   1  3.8407  0.051077
## overall_score_numeric_less_trimmed:country   2478   5  1.7568  0.122103
## Residuals                                   74201 263
##
## (Intercept)                                ***
## overall_score_numeric_less_trimmed         **
## country                                    .
## sex
## traveled_before                            .
## overall_score_numeric_less_trimmed:country
## Residuals
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

So we do have a significant `overall_score_numeric_trimmed:country` interaction, as well as a significant main effect of both variables. We note that this is not true for English confidence - only a main effect of English confidence (and previous travel):

```r
Anova(lm(stress ~ english_confidence * country + sex + traveled_before, data = noOtherCountry), type =
```

```
## Anova Table (Type III tests)
##
## Response: stress
##                            Sum Sq  Df  F value    Pr(>F)
## (Intercept)                 37919   1 147.9127 < 2.2e-16 ***
## english_confidence           3216   1  12.5446 0.0004551 ***
## country                      2333   5   1.8203 0.1083385
## sex                            39   1   0.1525 0.6964474
## traveled_before              1389   1   5.4193 0.0205245 *
## english_confidence:country   1670   5   1.3031 0.2622034
## Residuals                   83829 327
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Here are the marginal means for stress levels by country, followed by the estimates of the slopes for `overall_score_numeric_less_trimmed` by country:

```r
lsmeans(countryModel, ~ overall_score_numeric_less_trimmed * country)
```

```
##  overall_score_numeric_less_trimmed country                     lsmean
##                            6.837545 USA                       66.05463
##                            6.837545 Canada                    54.64908
##                            6.837545 UK / Ireland              58.43142
##                            6.837545 Australia / New Zealand   61.89670
```

```
##                             6.837545 Germany / France / Holland 60.84482
##                             6.837545 Korea / Japan / Singapore  61.53002
##        SE  df lower.CL upper.CL
##  1.960160 263 62.19503 69.91424
##  3.947871 263 46.87563 62.42254
##  2.245755 263 54.00947 62.85337
##  2.330855 263 57.30719 66.48621
##  4.112225 263 52.74774 68.94189
##  4.228466 263 53.20407 69.85598
##
## Results are averaged over the levels of: sex, traveled_before
## Confidence level used: 0.95
```

```
countryTrends <- lstrends(countryModel, ~ country, var = "overall_score_numeric_less_trimmed")
cld(countryTrends)
```

```
##  country                    overall_score_numeric_less_trimmed.trend
##  USA                                                      -7.3494892
##  UK / Ireland                                             -3.1537021
##  Australia / New Zealand                                  -0.8325595
##  Canada                                                   -0.6412997
##  Germany / France / Holland                                5.6090267
##  Korea / Japan / Singapore                                 6.3958836
##        SE  df   lower.CL  upper.CL .group
##  2.771901 263 -12.807432 -1.891546  1
##  2.909799 263  -8.883170  2.575766  1
##  2.833856 263  -6.412493  4.747374  1
##  8.242752 263 -16.871484 15.588885  1
##  4.864707 263  -3.969703 15.187756  1
##  5.628628 263  -4.687026 17.478793  1
##
## Results are averaged over the levels of: sex, traveled_before
## Confidence level used: 0.95
## P value adjustment: tukey method for comparing a family of 6 estimates
## significance level used: alpha = 0.05
```

So it looks like in the USA, UK, and Australia, a higher test score is associated with lower stress levels. This is interesting because these are all the countries, except Canada, where English is actually the native language. Students who study in Canada experience the lowest stress of the group, and there is essentially a null effect of test scores on stress levels in Canada (trend = -0.6, i.e., near zero), so there may be something special going on for this country. Also, the .group output shows us that the effect of test scores on stress levels for students in the USA is significantly different than the effect of test scores in Germany / France / Holland: whereas higher test scores are associated with lower stress in the USA (slope = -9.3, so for every 1 point increase in test scores, we expect 9.3 lower reported stress "level"), the opposite is true in Germany / France / Holland. This is sort of hard to explain! But an important insight from this analysis is that the effect of test scores does appear to be important within the US. Let's perform individual regressions on the various countries, and then plot a graph with regression lines by country. A linear mixed-effects approach was also conducted to capture heterogeneity by country, which is presented at the end of this document - this analysis produced similar results (though not all random effects of country were found to be statistically significant); it also indicated that countries with the highest (intercept) stress levels also showed the greatest improvement in stress with higher test scores, while those with the lowest stress levels were the ones where higher test scores were associated with higher stress.

```
countries <- unique(refiltered$country)
countries_list <- list()
countries_names <- character(0)
```

```r
counter <- 0
for(i in countries) {
  counter <- counter + 1
  countries_names[counter] <- i
  countries_list[[counter]] <- lm(stress ~ overall_score_numeric_less_trimmed + traveled_before + sex, d
  print(paste0("Country = ", i))
  print(summary(countries_list[[counter]]))
}
```

```
## [1] "Country = UK / Ireland"
##
## Call:
## lm(formula = stress ~ overall_score_numeric_less_trimmed + traveled_before +
##     sex, data = refiltered, subset = country == i)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -27.210 -12.210  -4.533   9.644  48.996
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                         83.5601    21.9577   3.806  0.00033
## overall_score_numeric_less_trimmed  -3.5884     3.0815  -1.165  0.24875
## traveled_beforeAbroad before        -0.7346     4.8573  -0.151  0.88028
## sexFemale                            0.5034     5.4237   0.093  0.92636
##
## (Intercept)                        ***
## overall_score_numeric_less_trimmed
## traveled_beforeAbroad before
## sexFemale
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17.4 on 61 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.02509,    Adjusted R-squared:  -0.02285
## F-statistic: 0.5234 on 3 and 61 DF,  p-value: 0.6678
##
## [1] "Country = Germany / France / Holland"
##
## Call:
## lm(formula = stress ~ overall_score_numeric_less_trimmed + traveled_before +
##     sex, data = refiltered, subset = country == i)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -30.676  -7.591  -0.050   6.907  35.659
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                          24.326     36.564   0.665    0.517
## overall_score_numeric_less_trimmed    4.611      5.024   0.918    0.374
## traveled_beforeAbroad before         10.158     10.929   0.929    0.368
## sexFemale                             6.045      9.571   0.632    0.538
```

```
##
## Residual standard error: 16.75 on 14 degrees of freedom
##   (26 observations deleted due to missingness)
## Multiple R-squared:  0.1276, Adjusted R-squared:  -0.05939
## F-statistic: 0.6823 on 3 and 14 DF,  p-value: 0.5774
##
## [1] "Country = Korea / Japan / Singapore"
##
## Call:
## lm(formula = stress ~ overall_score_numeric_less_trimmed + traveled_before +
##     sex, data = refiltered, subset = country == i)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -25.967  -9.989  -1.325  10.678  25.484
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                          25.637     39.889   0.643    0.530
## overall_score_numeric_less_trimmed    5.098      6.003   0.849    0.409
## traveled_beforeAbroad before         -9.053      8.784  -1.031    0.319
## sexFemale                             6.191      8.653   0.715    0.485
##
## Residual standard error: 17.37 on 15 degrees of freedom
##   (21 observations deleted due to missingness)
## Multiple R-squared:  0.1675, Adjusted R-squared:  0.001036
## F-statistic: 1.006 on 3 and 15 DF,  p-value: 0.4173
##
## [1] "Country = Canada"
##
## Call:
## lm(formula = stress ~ overall_score_numeric_less_trimmed + traveled_before +
##     sex, data = refiltered, subset = country == i)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -24.911  -7.039   1.452   5.610  19.349
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                         54.2609    42.8568   1.266   0.2226
## overall_score_numeric_less_trimmed   0.5215     6.4699   0.081   0.9367
## traveled_beforeAbroad before       -11.3629     6.1565  -1.846   0.0824 .
## sexFemale                            2.3888     5.8844   0.406   0.6898
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 12.95 on 17 degrees of freedom
##   (1 observation deleted due to missingness)
## Multiple R-squared:  0.1706, Adjusted R-squared:  0.02423
## F-statistic: 1.166 on 3 and 17 DF,  p-value: 0.3519
##
## [1] "Country = Australia / New Zealand"
##
```

```
## Call:
## lm(formula = stress ~ overall_score_numeric_less_trimmed + traveled_before +
##     sex, data = refiltered, subset = country == i)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -32.88 -11.58  -1.19  10.36  49.12
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                         67.7644    18.7568   3.613 0.000602
## overall_score_numeric_less_trimmed  -0.7149     2.9571  -0.242 0.809751
## traveled_beforeAbroad before       -10.1840     5.0657  -2.010 0.048676
## sexFemale                            3.7572     4.4474   0.845 0.401412
##
## (Intercept)                        ***
## overall_score_numeric_less_trimmed
## traveled_beforeAbroad before       *
## sexFemale
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.84 on 63 degrees of freedom
##   (6 observations deleted due to missingness)
## Multiple R-squared:  0.07586,    Adjusted R-squared:  0.03185
## F-statistic: 1.724 on 3 and 63 DF,  p-value: 0.1711
##
## [1] "Country = Other"
##
## Call:
## lm(formula = stress ~ overall_score_numeric_less_trimmed + traveled_before +
##     sex, data = refiltered, subset = country == i)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -18.140  -8.654  -3.635   3.784  41.176
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                          -23.11     136.55  -0.169    0.870
## overall_score_numeric_less_trimmed    11.37      18.96   0.600    0.565
## traveled_beforeAbroad before         -15.71      14.65  -1.072    0.315
## sexFemale                             18.03      21.38   0.844    0.423
##
## Residual standard error: 19.39 on 8 degrees of freedom
##   (5 observations deleted due to missingness)
## Multiple R-squared:  0.2042, Adjusted R-squared:  -0.09424
## F-statistic: 0.6842 on 3 and 8 DF,  p-value: 0.5863
##
## [1] "Country = USA"
##
## Call:
## lm(formula = stress ~ overall_score_numeric_less_trimmed + traveled_before +
##     sex, data = refiltered, subset = country == i)
```

```
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -32.929 -12.862   0.255  11.355  40.708
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                         119.623     19.944   5.998 5.01e-08
## overall_score_numeric_less_trimmed   -7.128      2.826  -2.522   0.0136
## traveled_beforeAbroad before         -2.363      3.714  -0.636   0.5265
## sexFemale                            -6.184      3.728  -1.659   0.1009
##
## (Intercept)                        ***
## overall_score_numeric_less_trimmed *
## traveled_beforeAbroad before
## sexFemale
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 17 on 83 degrees of freedom
##   (9 observations deleted due to missingness)
## Multiple R-squared:  0.1183, Adjusted R-squared:  0.08647
## F-statistic: 3.714 on 3 and 83 DF,  p-value: 0.01465
```

```
ggplot(refiltered, aes(x=overall_score_numeric_less_trimmed, y=stress, color=country)) + geom_jitter(wi
```

```
## Warning: Removed 69 rows containing non-finite values (stat_smooth).
```

```
## Warning: Removed 69 rows containing missing values (geom_point).
```

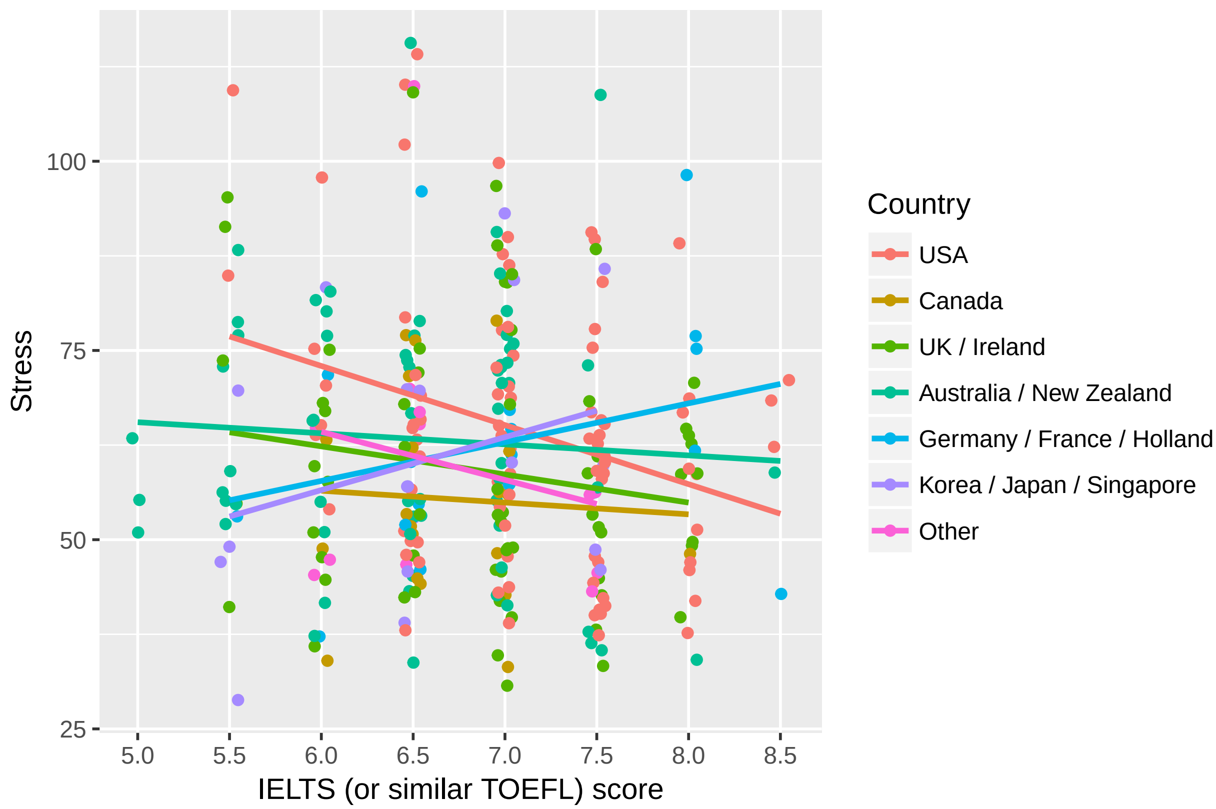


Figure 8: Relationship between English confidence and stress by country

Only in the USA - notably, the country with the highest average reported stress - did test scores significantly relate to stress levels, with a one point IELTS score increase being associated with a 7.128 point decrease in stress. Thus, it is possible that test scores are predictive of lower stress in settings where stress has a greater likelihood of being higher.

```
texreg(countries_list, type = "html",  digits = 3, bold = .05, booktabs = TRUE, sideways = TRUE, use.pa
```

Analyzing only countries where sample was large enough - first we graph the percent of people who took the exam by country

```
table(refiltered$country, refiltered$took_test)
```

```
##
##                               Did not take test Took test
##   USA                                         8        88
##   Canada                                      1        21
##   UK / Ireland                                1        65
##   Australia / New Zealand                     6        67
##   Germany / France / Holland                 26        18
##   Korea / Japan / Singapore                  20        20
##   Other                                       4        13
```

```
ggplot(refiltered, aes(x=country, fill=took_test)) + geom_bar(position="stack") + labs(x="Country", y="
```

	UK / Ireland	Germany / France / Holland	Korea / Japan / Singapore	Canada	Australia / New Zealand	Other	USA
(Constant)	**83.560*****	24.326	25.637	54.261	**67.764*****	−23.106	**119.623*****
	(21.958)	(36.564)	(39.889)	(42.857)	(18.757)	(136.551)	(19.944)
Test scores	−3.588	4.611	5.098	0.521	−0.715	11.369	**−7.128***
	(3.082)	(5.024)	(6.003)	(6.470)	(2.957)	(18.963)	(2.826)
Gender (Female)	0.503	6.045	6.191	2.389	3.757	18.033	−6.184
	(5.424)	(9.571)	(8.653)	(5.884)	(4.447)	(21.378)	(3.728)
Prior travel	−0.735	10.158	−9.053	−11.363	**−10.184***	−15.714	−2.363
	(4.857)	(10.929)	(8.784)	(6.156)	(5.066)	(14.655)	(3.714)
R^2	0.025	0.128	0.168	0.171	0.076	0.204	0.118
Adj. R^2	-0.023	-0.059	0.001	0.024	0.032	-0.094	0.086
Num. obs.	65	18	19	21	67	12	87

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$.
Results are as coefficient (standard error).

Table 5: Relationship between test scores and stress by country

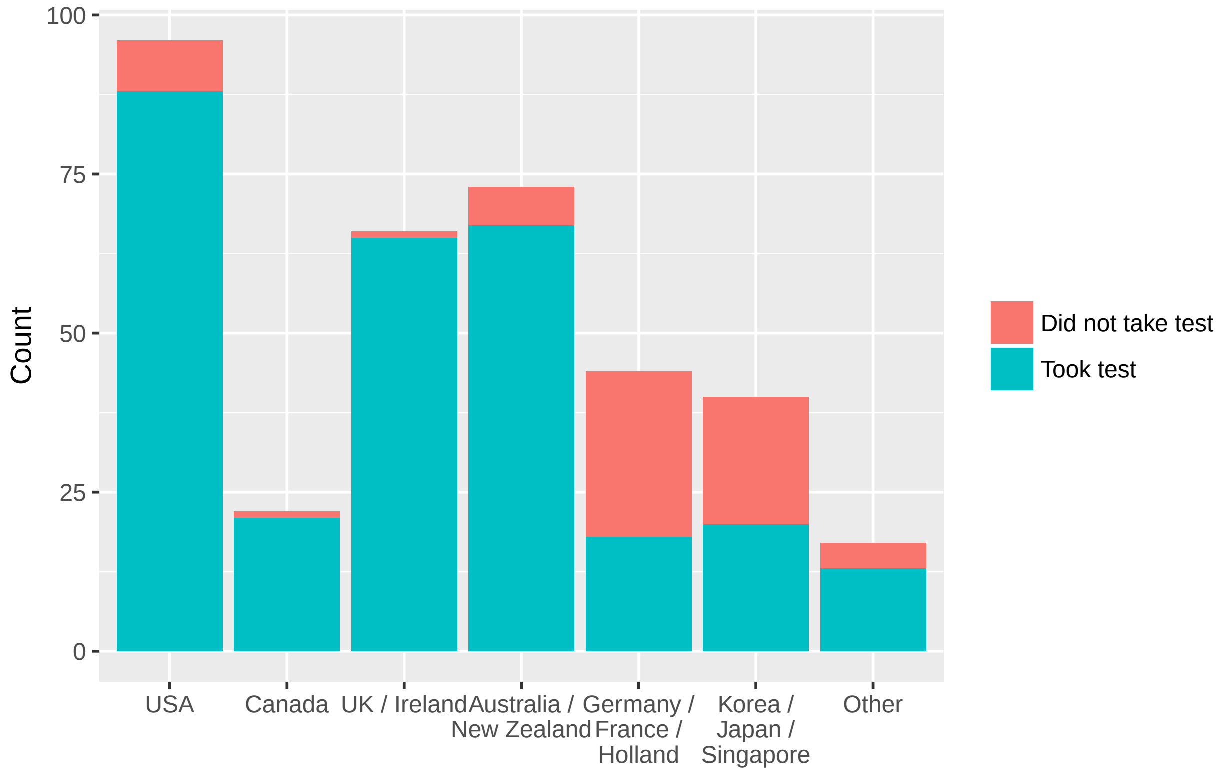


Figure 9: Number of subjects who did or did not take English exams by country

```
Anova(lm(stress ~ overall_score_numeric_less_trimmed * country + sex + traveled_before, data = refiltere
```

```
## Anova Table (Type III tests)
##
## Response: stress
##                                            Sum Sq  Df F value    Pr(>F)
## (Intercept)                                 10251   1 36.0481 6.116e-09
## overall_score_numeric_less_trimmed           1978   1  6.9561  0.008833
## country                                      2693   6  1.5783  0.153453
## sex                                             5   1  0.0173  0.895430
## traveled_before                              1268   1  4.4604  0.035597
## overall_score_numeric_less_trimmed:country   2508   6  1.4701  0.188464
## Residuals                                   77634 273
##
## (Intercept)                                ***
## overall_score_numeric_less_trimmed         **
## country
## sex
## traveled_before                            *
## overall_score_numeric_less_trimmed:country
## Residuals
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
refiltered <- refiltered %>% mutate(good_sample_countries = ifelse(country == "USA" | country == "UK / I
refiltered$good_sample_countries <- factor(refiltered$good_sample_countries, labels = c("Lower sample",
```

```r
good_vs_bad_countries <- lm(stress ~ overall_score_numeric_less_trimmed * good_sample_countries + sex +
summary(good_vs_bad_countries)
```

```
##
## Call:
## lm(formula = stress ~ overall_score_numeric_less_trimmed * good_sample_countries +
##     sex + traveled_before, data = refiltered)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -30.816 -12.954  -2.121   9.753  51.184
##
## Coefficients:
##                                                                Estimate
## (Intercept)                                                     36.2620
## overall_score_numeric_less_trimmed                               3.6942
## good_sample_countriesHigher sample                              49.5986
## sexFemale                                                       -0.6478
## traveled_beforeAbroad before                                    -4.4082
## overall_score_numeric_less_trimmed:good_sample_countriesHigher sample  -6.8321
##                                                                Std. Error
## (Intercept)                                                       20.6393
## overall_score_numeric_less_trimmed                                 3.0337
## good_sample_countriesHigher sample                                23.1912
## sexFemale                                                          2.1185
## traveled_beforeAbroad before                                       2.1853
## overall_score_numeric_less_trimmed:good_sample_countriesHigher sample     3.4160
##                                                                t value
## (Intercept)                                                      1.757
## overall_score_numeric_less_trimmed                               1.218
## good_sample_countriesHigher sample                               2.139
## sexFemale                                                       -0.306
## traveled_beforeAbroad before                                    -2.017
## overall_score_numeric_less_trimmed:good_sample_countriesHigher sample  -2.000
##                                                                Pr(>|t|)
## (Intercept)                                                      0.0800
## overall_score_numeric_less_trimmed                               0.2243
## good_sample_countriesHigher sample                               0.0333
## sexFemale                                                        0.7600
## traveled_beforeAbroad before                                     0.0446
## overall_score_numeric_less_trimmed:good_sample_countriesHigher sample   0.0465
##
## (Intercept)                                                    .
## overall_score_numeric_less_trimmed
## good_sample_countriesHigher sample                             *
## sexFemale
## traveled_beforeAbroad before                                   *
## overall_score_numeric_less_trimmed:good_sample_countriesHigher sample *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.92 on 283 degrees of freedom
##   (69 observations deleted due to missingness)
## Multiple R-squared:  0.04436,    Adjusted R-squared:  0.02747
```

```
## F-statistic: 2.627 on 5 and 283 DF,  p-value: 0.02427
```

```
lsmeans(good_vs_bad_countries, ~ overall_score_numeric_less_trimmed * good_sample_countries)
```

```
##  overall_score_numeric_less_trimmed good_sample_countries   lsmean
##                             6.83218 Lower sample          58.97355
##                             6.83218 Higher sample         61.89381
##        SE  df lower.CL upper.CL
##  2.113596 283 54.81318 63.13391
##  1.257139 283 59.41928 64.36834
##
## Results are averaged over the levels of: sex, traveled_before
## Confidence level used: 0.95
```

```
nativeTrends <- lstrends(good_vs_bad_countries, ~ good_sample_countries, var = "overall_score_numeric_le
cld(nativeTrends)
```

```
##  good_sample_countries overall_score_numeric_less_trimmed.trend       SE
##  Higher sample                                        -3.137912 1.584039
##  Lower sample                                          3.694214 3.033725
##   df  lower.CL   upper.CL .group
##  283 -6.255906 -0.0199183  1
##  283 -2.277314  9.6657431   2
##
## Results are averaged over the levels of: sex, traveled_before
## Confidence level used: 0.95
## significance level used: alpha = 0.05
```

```
only_good_countries <- lm(stress ~ overall_score_numeric_less_trimmed + sex + traveled_before, data = re
only_bad_countries <- lm(stress ~ overall_score_numeric_less_trimmed + sex + traveled_before, data = ref
split_countries_model_list = list(good_vs_bad_countries, only_good_countries, only_bad_countries)
```

Outputting a table of the results from this approach

```
texreg(split_countries_model_list, type = "html",  digits = 3, bold = .05, booktabs = TRUE, sideways = F
```

Mediation model: scores, confidence, and stress

Returning to the idea that test scores are correlated with English confidence but to not show a main effect on stress alone but rather show an effect on stress that is moderated by country, perhaps we an explain this further. Test scores are an 'objective' measure, while English confidence comprised of a series of subjective self-report Likert-style qestions. Importantly, the stress measure is also a subjective self-report scale, so perhaps it is not so surprising that there appear to be a more evident direct relationship between English confidence and stress - i.e., someone who is engaging in adaptive coping strategies, feels 'good', might tend to report higher subjective confidence in their language and correspondingly lower stress. Perhaps the fact that both of these are self-reported subjective measures explains their greater correspondence, which could explain why this relationship does not appear to be dependent on the country where one is studying. That is, the correspondence between psychological measures would not necessarily be expected to depend on country of residence. However, our correlation test did indicate that test scores are correlated with English confidence. Perhaps this objective measure of English ability is abstracted into the psychological measure of English confidence, which ultimately mediates the predictive relationship between test scores and stress. We'll test this directly using a mediation analysis: we will test whether there is a statistically significant indirect effect of test scores (or association with test scores) on stress acting through English confidence - whether English confidence mediates the relationship between test scores and stress.

		High sample	Low sample
(Constant)	36.262	**87.320*****	29.067
	(20.639)	(11.092)	(19.611)
Test scores	3.694	$-$**3.235***	4.417
	(3.034)	(1.627)	(2.830)
High sample	**49.599***		
	(23.191)		
Test scores x high sample	$-$**6.832***		
	(3.416)		
Gender (Female)	-0.648	-2.401	4.388
	(2.118)	(2.489)	(3.966)
Prior travel	$-$**4.408***	-3.248	-7.576
	(2.185)	(2.551)	(4.178)
R^2	0.044	0.037	0.091
Adj. R^2	0.027	0.024	0.049
Num. obs.	289	219	70

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$.
Results are as coefficient (standard error).

Table 6: The relationship of test scores and stress, segregated by country sample size

```
m <- lm(english_confidence ~ overall_score_numeric_less_trimmed + sex + traveled_before, data = refilte
y <- lm(stress ~ english_confidence + overall_score_numeric_less_trimmed + sex + traveled_before, data =
med_scores_conf_stress <- mediate(m, y, sims = 100, boot = TRUE, mediator = "english_confidence", treat
summary(med_scores_conf_stress)
```

```
##
## Causal Mediation Analysis
##
## Nonparametric Bootstrap Confidence Intervals with the Percentile Method
##
##                 Estimate 95% CI Lower 95% CI Upper p-value
## ACME              -1.661       -2.434        -0.68  <2e-16 ***
## ADE                0.133       -2.720         2.40    0.60
## Total Effect      -1.528       -3.670         0.89    0.24
## Prop. Mediated     1.087      -12.764        18.71    0.24
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Sample Size Used: 289
##
##
## Simulations: 100
```

```
texreg(list(m,y), type = "html",  digits = 3, bold = .05, booktabs = TRUE, sideways = FALSE, use.packag
```

What this tells is is that there is a significant Average Causal Mediated Effect (ACME) but no significant Average Direct Effect (ADE). The ACME is calculated as `a*b` where `a` is derived from the mediation regression equation `Y(mediator: English confidence) = a*test score + X*covariates + e`, and `b` is derived from the equation `Y(stress) = b*English confidence + c'*test_score + X*covariates + e`. `c'` is the ADE. The ACME test statistic is calculated by bootstrap resampling. Thus it appears that the association between test scores and stress levels is mediated by English confidence, i.e., it has an indirect effect on test scores that is mediated through its effect on English confidence. The absence of a significant

	Mediator: English confidence	Stress
(Constant)	**3.559***	**78.167*****
	(1.617)	(9.599)
a, c': Test scores	**1.504*****	0.133
	(0.236)	(1.487)
b: English confidence		$-$**1.104****
		(0.349)
Gender (Female)	0.275	-0.653
	(0.355)	(2.094)
Prior travel	0.231	$-$**4.316***
	(0.367)	(2.165)
R^2	0.136	0.057
Adj. R^2	0.127	0.043
Num. obs.	289	289

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$.
Results are as coefficient (standard error).

Table 7: English confidence mediates the relationship between test scores and stress. ab = -1.661. 95% confidence interval: [-2.88, -0.57], p = 0.002

ADE indicates 'full' mediation - once the mediator is controlled for, there's no significant effect of test scores on stress. Note that because this study did not perform experimental manipulation we cannot interpret our mediated effects as causal, but rather only associative.

This result is consistent with the hypothesis that higher test scores are associated with higher self-appraisal of English ability (English confidence), which in turn is associated with lower levels of self-appraised stress (or, correspondingly, higher degree of self-appraised well-being).

More mediation models

The relationship between test scores, English confidence, and stress is interesting, but perhaps we can dig a little deeper. It makes sense that an objective measure of English ability such as IELTS / TOEFL would be abstracted into a psychological measure - English confidence - in order to be associated with the psychological measure of stress. But is there another factor that mediates the relationship between English confidence and stress? How does English confidence translate into lower stress levels - through potential coping mechanisms? One possible factor is that it is associated with greater social interaction, which in turn ameliorates stress levels.

First let's see if English confidence is predictive of likelihood / frequency of interaction with native friends, as measured on a 5-point Likert scale. This dependent measure is an ordinal scale variable so we'll use an ordered logit model. We'll also plot the data with a linear regression line for reference.

```
refiltered$friends_factor <- factor(refiltered$native_friends)
friends <- polr(friends_factor ~ english_confidence + sex + traveled_before, data = refiltered, Hess = '
coefs <- coef(summary(friends))
ps <- pnorm(abs(coefs[,'t value']), lower.tail = FALSE) * 2
ptable <- cbind(coefs, "p value" = ps)
ptable
```

```
##                                  Value Std. Error   t value      p value
## english_confidence           0.2494812 0.03356325 7.4331679 1.060272e-13
## sexFemale                    0.1071153 0.20012421 0.5352442 5.924810e-01
## traveled_beforeAbroad before 0.2899027 0.21121069 1.3725760 1.698842e-01
## 1|2                          1.7029766 0.46434792 3.6674583 2.449734e-04
```

```
## 2|3                          2.8395530 0.47083143 6.0309333 1.630154e-09
## 3|4                          4.1203471 0.49686704 8.2926553 1.107479e-16
## 4|5                          4.7617233 0.51300324 9.2820532 1.662436e-20
```

```
ggplot(refiltered, aes(x=friends_factor, y=english_confidence)) + geom_boxplot() +  labs(x="Degree of in
```

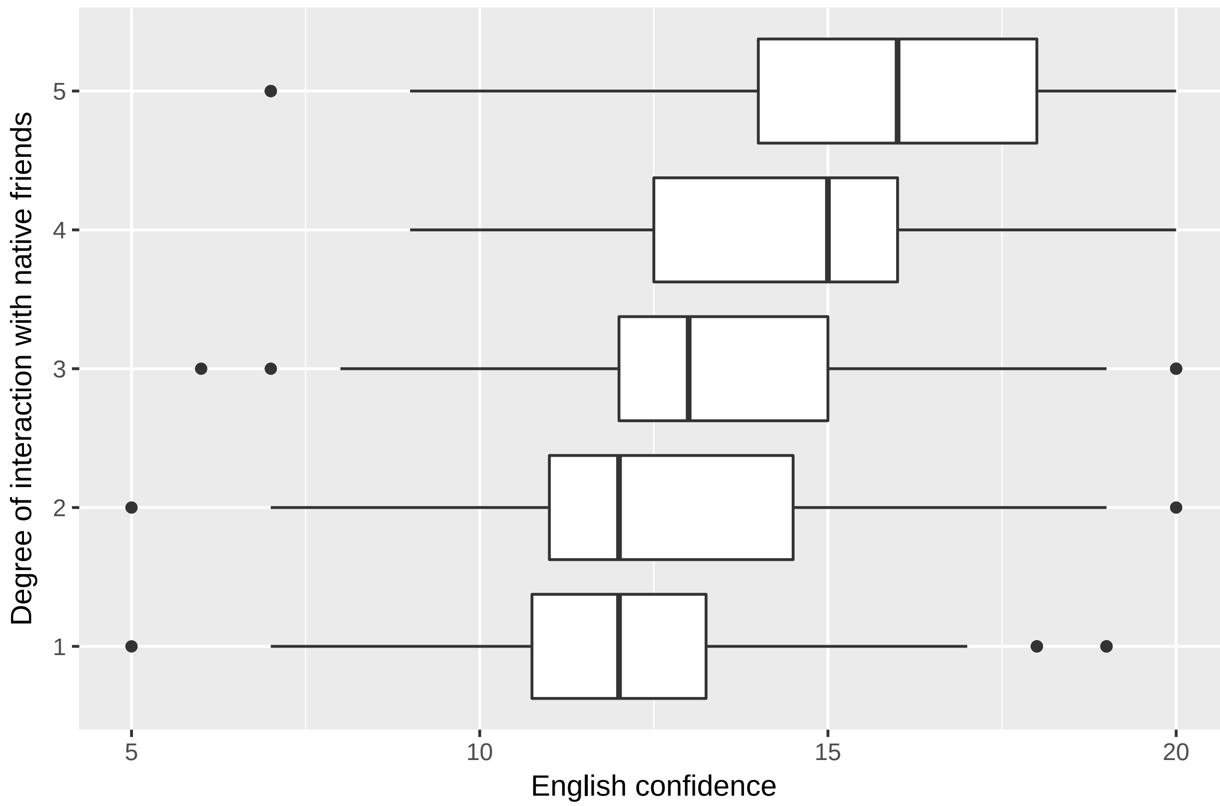


Figure 10: Higher English confidence is associated with greater interation with native friends

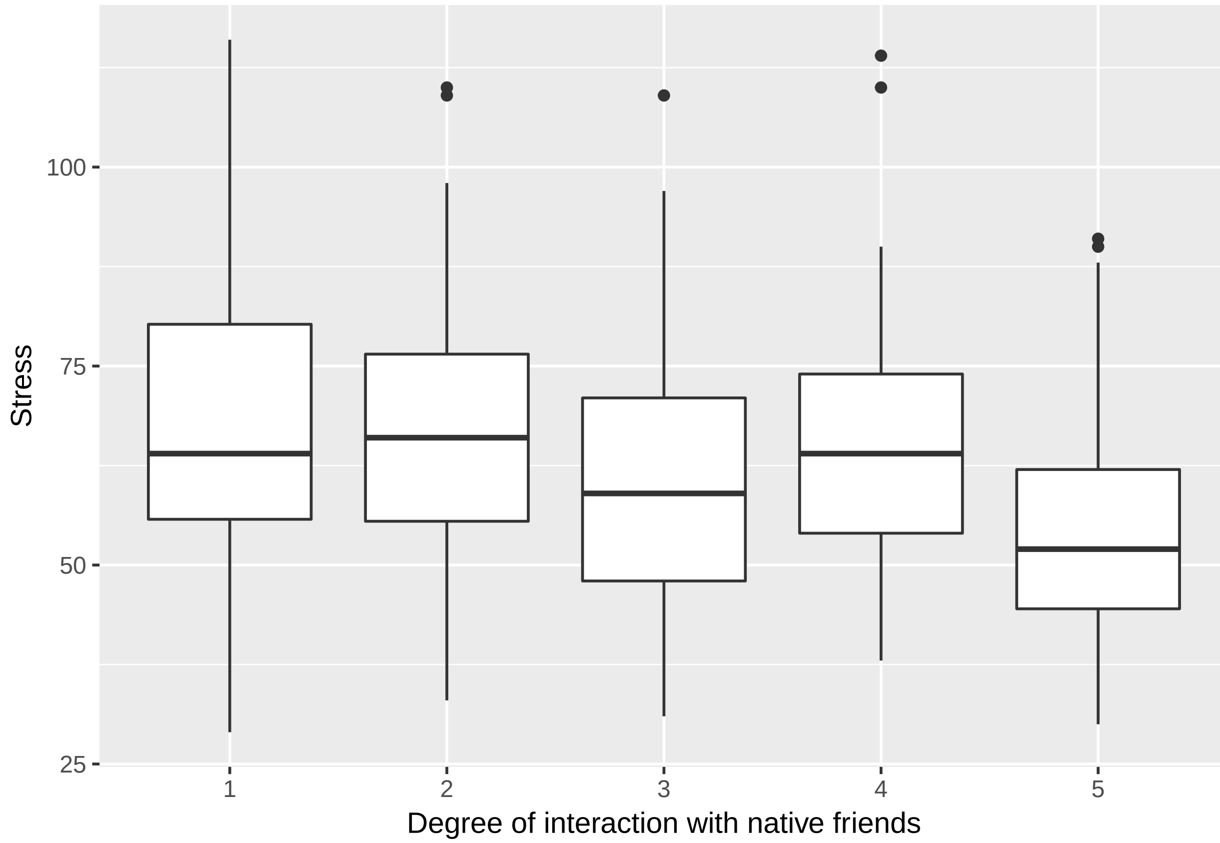
So it looks like there's a significant relationship between English confidence and social interaction with natives - a 1 unit increase in English confidence is associated with .25 log odds increase in probability of a higher reported value on the likert scale of social interaction. Next we'll see if social interaction is related to stress levels.

```
Anova(lm(stress ~ friends_factor + traveled_before + sex, data = refiltered), type = "III")
```

```
## Anova Table (Type III tests)
##
## Response: stress
##                 Sum Sq  Df  F value    Pr(>F)
## (Intercept)     189861   1 763.4993 < 2.2e-16 ***
## friends_factor    8787   4   8.8338 8.335e-07 ***
## traveled_before    723   1   2.9061   0.08913 .
## sex                 31   1   0.1230   0.72606
## Residuals        87284 351
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Looks like interaction with friends is very significantly predictive of stress levels. Let's plot the relationship.

```r
ggplot(refiltered, aes(x=friends_factor, y=stress)) + geom_boxplot() + labs(x="Degree of interaction wi
```



We now have pretty good reason to believe social interaction with friends might mediate the relationship between English confidence and stress levels. Let's run the mediation analysis.

```r
m <- friends
y <- lm(stress ~ friends_factor + english_confidence + sex + traveled_before, data=refiltered)
med_conf_friends_stress <- mediate(m, y, sims = 100, boot = TRUE, mediator = "friends_factor", treat =
summary(med_conf_friends_stress)
```

```
##
## Causal Mediation Analysis
##
## Nonparametric Bootstrap Confidence Intervals with the Percentile Method
##
##                  Estimate 95% CI Lower 95% CI Upper p-value
## ACME               -0.578       -1.107         0.01    0.06 .
## ADE                -0.330       -0.849         0.14    0.20
## Total Effect       -0.908       -1.459        -0.18    0.02 *
## Prop. Mediated      0.636       -0.407         1.25    0.08 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Sample Size Used: 358
##
##
## Simulations: 100
```

The mediation analysis using the ordered logit model doesn't correspond well with our intuitive sense of the data, with English confidence related to social interaction, which is related to stress. Indeed, adding social interaction to the model predicting stress makes English confidence no longer significant, strongly suggesting a mediated effect:

```
Anova(y, type = "III")
```

```
## Anova Table (Type III tests)
##
## Response: stress
##                     Sum Sq  Df  F value    Pr(>F)
## (Intercept)          70602   1 284.1649 < 2.2e-16 ***
## friends_factor        6726   4   6.7682 2.951e-05 ***
## english_confidence     325   1   1.3089    0.2534
## sex                     29   1   0.1150    0.7347
## traveled_before        651   1   2.6193    0.1065
## Residuals            86959 350
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

Perhaps this is due to our use of the ordered logit model, meaning the coefficients don't combine well to form the `a*b` indirect effect in the mediation analysis, because one is on the log-odds scale (the model with the mediator as the dependent measure) and one is on the linear scale (the model with stress as the dependent measure). Let's refit the former on a linear scale - as is sometimes done when analyzing a likert scale - and see how our mediation analysis turns out. We'll repolot the relationship between social interaction and stress as a linear trend for good measure.

```
m <- lm(native_friends ~ english_confidence + sex + traveled_before, data = refiltered)
summary(m)
```

```
##
## Call:
## lm(formula = native_friends ~ english_confidence + sex + traveled_before,
##     data = refiltered)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -2.99380 -0.93707  0.04039  1.02438  3.07458
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    0.60977    0.31806   1.917    0.056 .
## english_confidence             0.17236    0.02161   7.975 2.14e-14 ***
## sexFemale                      0.10910    0.14527   0.751    0.453
## traveled_beforeAbroad before  0.25892    0.14941   1.733    0.084 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.288 on 354 degrees of freedom
## Multiple R-squared:  0.1684, Adjusted R-squared:  0.1614
## F-statistic:  23.9 on 3 and 354 DF,  p-value: 4.151e-14
```

```
y <- lm(stress ~ native_friends + english_confidence + sex + traveled_before, data = refiltered)
med_conf_friends_stress_linear <- mediate(m, y, sims = 100, boot = TRUE, mediator = "native_friends", t
summary(med_conf_friends_stress_linear)
```

```
##
```

```
## Causal Mediation Analysis
##
## Nonparametric Bootstrap Confidence Intervals with the Percentile Method
##
##                 Estimate 95% CI Lower 95% CI Upper p-value
## ACME              -0.484       -0.765        -0.27  <2e-16 ***
## ADE               -0.336       -0.857         0.19    0.14
## Total Effect      -0.820       -1.321        -0.36  <2e-16 ***
## Prop. Mediated     0.590        0.276         1.48  <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Sample Size Used: 358
##
##
## Simulations: 100
```

```
ggplot(refiltered, aes(x=friends_factor, y=stress)) + geom_boxplot() + geom_smooth(method="lm") + labs(
```

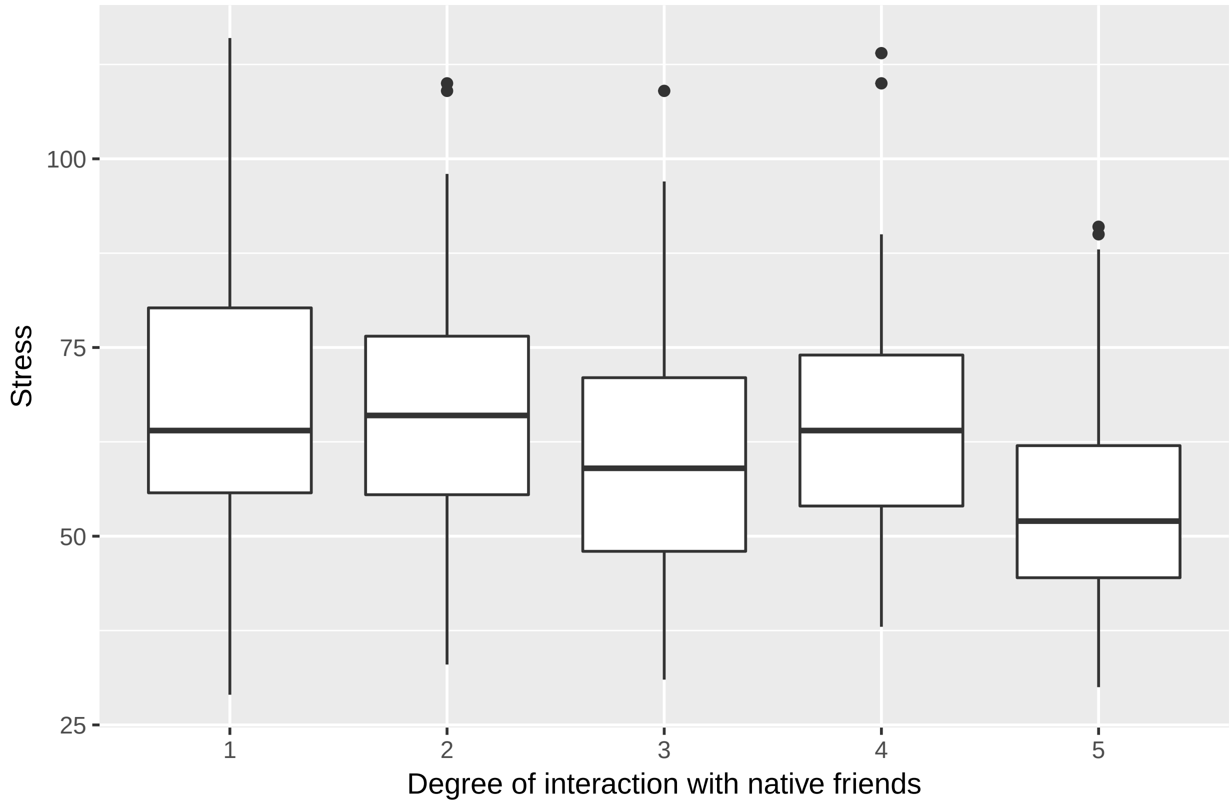


Figure 11: Higher interaction with native friends is associated with lower stress

So using the linear models we have evidence of an ACME: the relationship between English confidence and stress is mediated by degree of social interaction with native friends. Finally, let's output this model to a table.

```
texreg(list(m,y), type = "html",  digits = 3, bold = .05, booktabs = TRUE, sideways = FALSE, use.packag
```

	Mediator: Native friends	Stress
(Constant)	0.610	**77.206*****
	(0.318)	(3.944)
a, *c'*: English confidence	**0.172*****	−0.336
	(0.022)	(0.290)
b: Native friends		−**2.806*****
		(0.656)
Gender (Female)	0.109	−1.588
	(0.145)	(1.793)
Prior travel	0.259	−2.935
	(0.149)	(1.851)
R^2	0.168	0.089
Adj. R^2	0.161	0.079
Num. obs.	358	358

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$.
Results are as coefficient (standard error).

Table 8: Interaction with native friends mediates the relationship between English confidence and stress. *ab* = -0.484. 95% confidence interval: [-0.76, -0.24], $p < 10^{-16}$

Multiple mediator model

We now have 2 conceptual mediation models:

- Test scores -> English confidence -> stress
- English confidence -> native friends -> stress

What we want to do now is see if we can combine these models, using a multi-mediator approach:

- Test scores -> English confidence -> native friends -> stress

That is, we want to test if the effect of test scores is mediated by English confidence, the effect of which is mediated by social interaction, in a single statistical model that accounts for all possible sub-model paths such as:

- Test scores -> stress
- Test scores -> English confidence -> stress
- Test scores -> English confidence -> native friends -> stress
- Test scores -> native friends -> stress

We can't perform multiple serial/parallel mediation analysis using a continuous treatment variable as ours, test scores, is using the `medation` package in R, so we'll export the data frame to SPSS. We'll rename a few variables first to make our output in SPSS more readable because the PROCESS macro in SPSS does not allow for long variable names.

```
#refiltered %>% rename(confidence = english_confidence, test_scores_less_trimmed_numeric = overall_scor
```

The output of these analyses are found in multiple_mediators.htm. Bootstrap estimation of confidence intervals indicates that the the paths in **bold** below are statistically significant:

- Test scores -> stress
- Test scores -> English confidence -> stress

	Mediator: English confidence	Mediator: Native friends	Stress
(Constant)	**3.559***	-1.159	**75.396*****
	(1.617)	(0.719)	(9.503)
a_1, a_2, c': Test scores	**1.504*****	**0.242***	0.713
	(0.236)	(0.111)	(1.478)
d_{21}, b_1: English confidence		**0.180*****	-0.674
		(0.026)	(0.371)
b_2: Native friends			$-$**2.392****
			(0.780)
Gender (Female)	0.275	0.089	-0.442
	(0.355)	(0.157)	(2.065)
Prior travel	0.231	0.255	-3.706
	(0.367)	(0.162)	(2.143)
R^2	0.136	0.221	0.087
Adj. R^2	0.127	0.210	0.071
Num. obs.	289	289	289

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$.
Results are as coefficient (standard error).

Table 9: English confidence and interaction with native friends in conjunction mediate the relationship between test scores and stress. English confidence: ab = -1.01, 95% confidence interval: [-2.34, 0.004]. Native friends: ab = -.58, 95% confidence interval: [-1.51, -0.06]. Serial mediation: $a_1 d_{21} b_2$ = -0.65, 95% confidence interval: [-1.28, -.24]

- **Test scores -> English confidence -> native friends -> stress**
- **Test scores -> native friends -> stress**

Thus, there is no direct effect of test scores on stress, but test sccores act indirectly on stress through its influence on engagement with native friends, and through its influence on English confidence which in turn afffects native friendships which in serial affects stress. Both paths have negative coefficients, such that higher test scores are associated with lower stress indirectly through mediator variables.

We will use R to generate the associated regression output table in LaTeX format though.

```
m1_confidence <- lm(english_confidence ~ overall_score_numeric_less_trimmed + sex + traveled_before, dat
m2_friends <- lm(native_friends ~ overall_score_numeric_less_trimmed + english_confidence + sex + travel
y <- lm(stress ~ overall_score_numeric_less_trimmed + english_confidence + native_friends + sex + travel
```

```
texreg(list(m1_confidence, m2_friends, y), type = "html",  digits = 3, bold = .05, booktabs = TRUE, side
```

Secondary analyses

Here we analyze whether stress levels and English confidence are predictive of sexual experiences with either a person native to the country where the student studies abroad (`sex_native`) or with a Chinese person while studying abroad (`sex_chinese`), using 2 logistic regressions. We then ask whether sexual experiences are predictive of stress levels.

```
refiltered$had_sex <- refiltered$sex_native == "Yes" | refiltered$sex_chinese == "Yes"
refiltered$had_sex <- factor(refiltered$had_sex, labels = c('No', 'Yes'))
sex_native_glm <- glm(sex_native ~  english_confidence + sex + traveled_before, family="binomial", data
summary(sex_native_glm)
```

```
##
```

```
## Call:
## glm(formula = sex_native ~ english_confidence + sex + traveled_before,
##     family = "binomial", data = refiltered)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.2568  -0.5755  -0.3924  -0.2409   2.5425
##
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                  -6.47912    0.95984  -6.750 1.48e-11 ***
## english_confidence            0.33320    0.05944   5.606 2.07e-08 ***
## sexFemale                    -0.37799    0.33816  -1.118    0.264
## traveled_beforeAbroad before -0.23323    0.35412  -0.659    0.510
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 289.52  on 357  degrees of freedom
## Residual deviance: 250.44  on 354  degrees of freedom
## AIC: 258.44
##
## Number of Fisher Scoring iterations: 5
```

```
sex_chinese_glm <- glm(sex_chinese ~  english_confidence + sex + traveled_before, family="binomial", da
summary(sex_chinese_glm)
```

```
##
## Call:
## glm(formula = sex_chinese ~ english_confidence + sex + traveled_before,
##     family = "binomial", data = refiltered)
##
## Deviance Residuals:
##     Min       1Q   Median       3Q      Max
## -1.1913  -0.6472  -0.5623  -0.4754   2.1788
##
## Coefficients:
##                              Estimate Std. Error z value Pr(>|z|)
## (Intercept)                  -1.19811    0.60307  -1.987    0.047 *
## english_confidence            0.06154    0.04150   1.483    0.138
## sexFemale                    -1.30508    0.26273  -4.967 6.79e-07 ***
## traveled_beforeAbroad before -0.26500    0.29094  -0.911    0.362
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 387.71  on 357  degrees of freedom
## Residual deviance: 360.68  on 354  degrees of freedom
## AIC: 368.68
##
## Number of Fisher Scoring iterations: 4
```

```r
stress_sex <- lm(stress ~ sex_native + sex_chinese + sex + traveled_before, data = refiltered)
summary(stress_sex)
```

```
##
## Call:
## lm(formula = stress ~ sex_native + sex_chinese + sex + traveled_before,
##     data = refiltered)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -32.175 -12.020  -1.973  10.258  53.825
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    64.0328     1.8784  34.088   <2e-16 ***
## sex_nativeYes                   0.4182     2.5561   0.164   0.8701
## sex_chineseYes                  0.8801     2.1778   0.404   0.6864
## sexFemale                      -1.8576     1.9299  -0.963   0.3364
## traveled_beforeAbroad before  -4.2966     1.9018  -2.259   0.0245 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.49 on 353 degrees of freedom
## Multiple R-squared:  0.01806,    Adjusted R-squared:  0.006938
## F-statistic: 1.624 on 4 and 353 DF,  p-value: 0.1678
```

```r
stress_any_sex <- lm(stress ~ had_sex + sex + traveled_before, data = refiltered)
summary(stress_any_sex)
```

```
##
## Call:
## lm(formula = stress ~ had_sex + sex + traveled_before, data = refiltered)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -32.299 -11.979  -1.941  10.412  53.701
##
## Coefficients:
##                               Estimate Std. Error t value Pr(>|t|)
## (Intercept)                    64.3260     1.8596  34.592   <2e-16 ***
## had_sexYes                      0.2888     1.9165   0.151   0.8803
## sexFemale                      -2.0270     1.8951  -1.070   0.2855
## traveled_beforeAbroad before  -4.3200     1.8990  -2.275   0.0235 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.47 on 354 degrees of freedom
## Multiple R-squared:  0.01752,    Adjusted R-squared:  0.00919
## F-statistic: 2.104 on 3 and 354 DF,  p-value: 0.09943
```

```r
refiltered$prediction_sex_native <- predict(sex_native_glm, newdata = refiltered, type = "response")
ggplot(refiltered, aes(x=english_confidence, y=as.numeric(sex_native)-1)) +
  geom_jitter(width = .1, height = .1, alpha = .3) +
  geom_line(aes(x=english_confidence, y=prediction_sex_native), color = "blue", lwd = 1, lineend="round
  scale_y_discrete(limits = c(0,1), labels = c('No (0)', 'Yes (1)')) +
```

```r
  theme(aspect.ratio = .4, plot.caption=element_text(hjust=0)) +
  labs(x = "English confidence", y = "Sex: native", caption = "Figure 13: English confidence predicts li
```

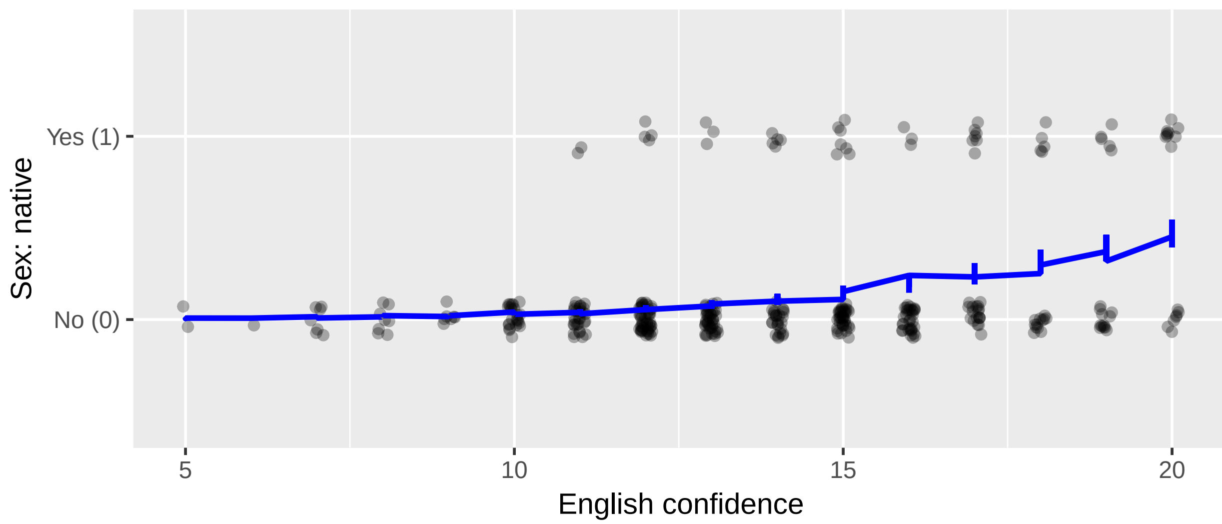


Figure 13: English confidence predicts likelihood of reporting sexual experience with native

```r
texreg(list(sex_native_glm, sex_chinese_glm, stress_sex), type = "html",  digits = 3, bold = .05, bookta
```

The results indicate that English confidence, but not gender, is predictive of sexual experiences with natives (1 unit increase in English confidence is associated with a .35 log-odds increase in likelihood to report a sexual experience), while gender, but not English confidence, is predicitve of sexual experiences with Chinese (females are 1.30 log odds less likely to report a sexual experience). These results make fairly good intuitive sense. Sexual experiences with neither natives nor Chinese are predictive of stress levels.

A mediation analysis, shown below, shows only a direct effect of English confidence on stress, i.e., this effect is not mediated by sexual experiences with natives.

```r
refiltered$sex_native_numeric <- as.numeric(refiltered$sex_native) - 1
m <- glm(sex_native_numeric ~ english_confidence + sex, family = "binomial", data = refiltered)
y <- lm(stress ~ english_confidence + sex_native_numeric + sex, data = refiltered)
med_conf_sex_stress <- mediate(m, y, sims = 100, boot = TRUE, mediator = "sex_native_numeric", treat =
summary(med_conf_sex_stress)
```

```
##
## Causal Mediation Analysis
##
## Nonparametric Bootstrap Confidence Intervals with the Percentile Method
##
##                  Estimate 95% CI Lower 95% CI Upper p-value
## ACME               0.1172      -0.0882         0.31    0.42
```

	Sex: native	Sex: Chinese	Stress
(Constant)	$\mathbf{-6.479}^{***}$	$\mathbf{-1.198}^{*}$	$\mathbf{64.033}^{***}$
	(0.960)	(0.603)	(1.878)
English confidence	$\mathbf{0.333}^{***}$	0.062	
	(0.059)	(0.041)	
Sex: native			0.418
			(2.556)
Sex: Chinese			0.880
			(2.178)
Gender (Female)	−0.378	$\mathbf{-1.305}^{***}$	−1.858
	(0.338)	(0.263)	(1.930)
Prior travel	−0.233	−0.265	$\mathbf{-4.297}^{*}$
	(0.354)	(0.291)	(1.902)
BIC	273.958	384.202	
Num. obs.	358	358	358
R^2			0.018
Adj. R^2			0.007

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$.
Results are as coefficient (standard error).

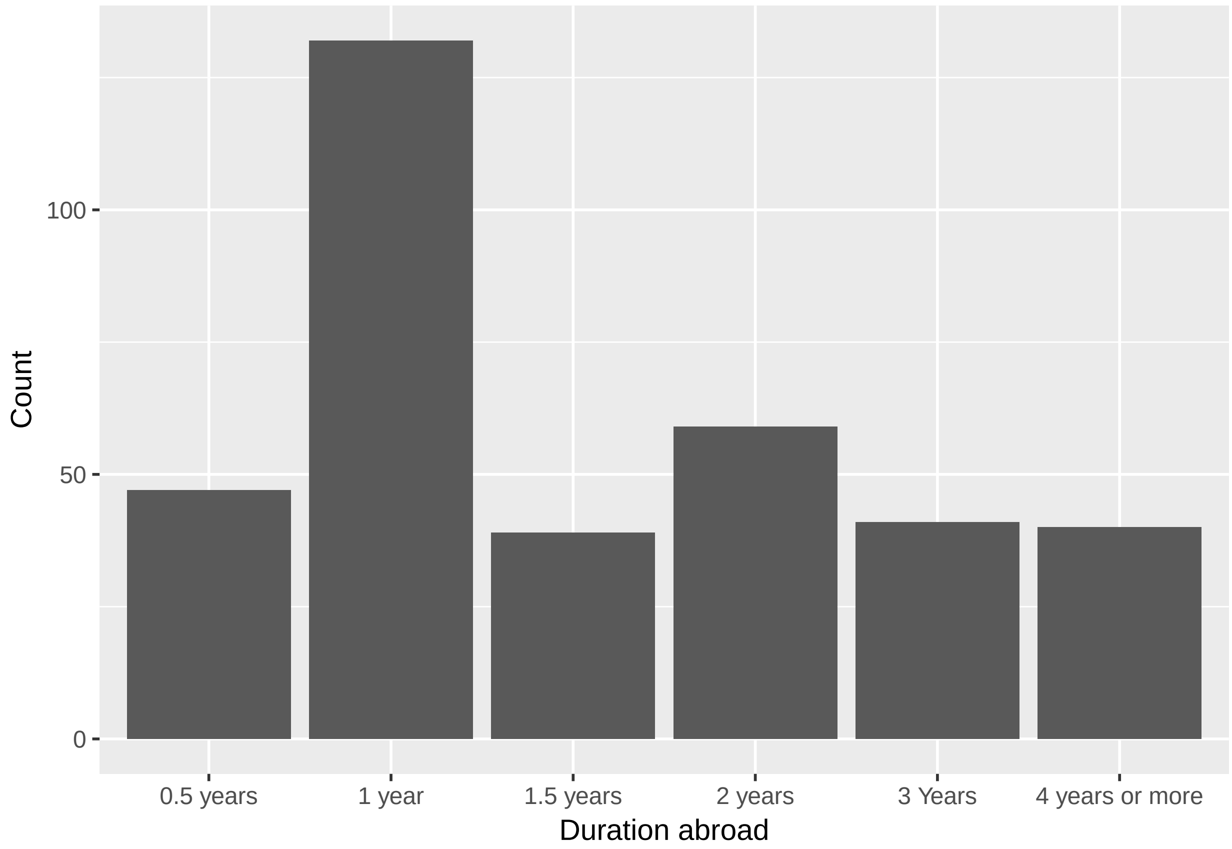
Table 10: The effect of English confidence on sexual experiences and sexual experiences on stress

```
## ADE              -1.0021      -1.4864        -0.58  <2e-16 ***
## Total Effect     -0.8850      -1.4017        -0.46  <2e-16 ***
## Prop. Mediated   -0.1324      -0.5182         0.08    0.42
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Sample Size Used: 358
##
##
## Simulations: 100
```

We'll next analyze whether duration abroad is associated with stress levels. First we'll plot the distribution of durations, followed by the stress levels by duration of stay.

```
ggplot(refiltered, aes(x=duration)) + geom_histogram(stat="count") + labs(x="Duration abroad", y="Count
```

```
## Warning: Ignoring unknown parameters: binwidth, bins, pad
```




```
ggplot(refiltered, aes(x=duration, y=stress)) + geom_boxplot() + labs(x="Duration abroad", y="Stress", 
```

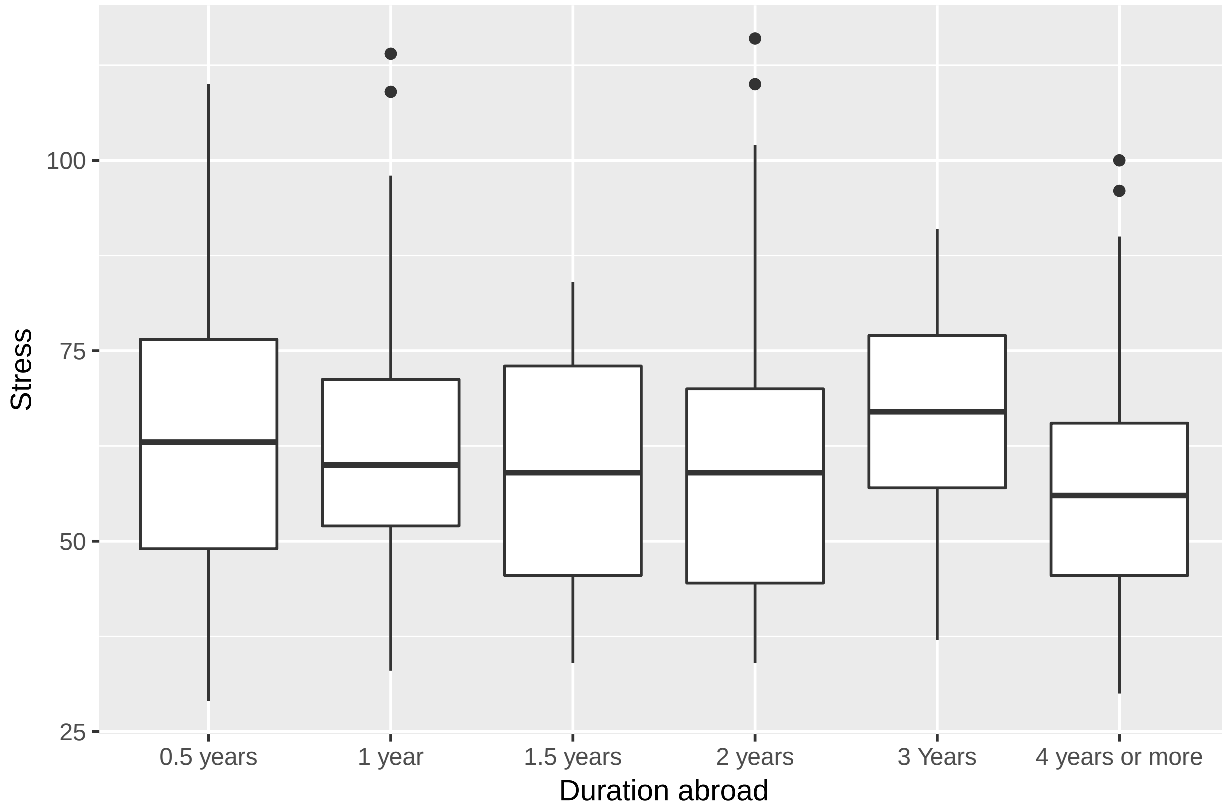


Figure 14: No relationship between duration abroad and stress

THere's no clear relationship between duration abroad and stress levels evident in this graph. We'll run the regressions to confirm this.

```
Anova(lm(stress ~ duration + sex + traveled_before, data = refiltered), type = "III")
```

```
## Anova Table (Type III tests)
##
## Response: stress
##                  Sum Sq  Df   F value  Pr(>F)
## (Intercept)      387661   1 1452.9433 < 2e-16 ***
## duration           2687   5    2.0143 0.07604 .
## sex                 200   1    0.7500 0.38705
## traveled_before    1363   1    5.1081 0.02443 *
## Residuals         93384 350
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The effect of duration is actually pretty close to statistically significant, which is somewhat surprising. Let's treat the `duration` as an numeric variable, `duration_numeric`, to see if there's a linear trend.

```
duration_model <- lm(stress ~ duration_numeric + sex + traveled_before, data = refiltered)
summary(duration_model)
```

```
##
## Call:
## lm(formula = stress ~ duration_numeric + sex + traveled_before,
##     data = refiltered)
##
```

	Stress
(Constant)	**65.673*****
	(2.182)
Duration abroad	−0.680
	(0.804)
Gender (Female)	−2.169
	(1.858)
Prior travel	−**4.296***
	(1.897)
R^2	0.019
Adj. R^2	0.011
Num. obs.	358

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$.
Results are as coefficient (standard error).

Table 11: No relationship between duration abroad and stress was found

```
## Residuals:
##     Min      1Q  Median      3Q     Max
## -31.993 -11.932  -1.993  10.408  53.856
##
## Coefficients:
##                              Estimate Std. Error t value Pr(>|t|)
## (Intercept)                   65.6727     2.1823  30.094   <2e-16 ***
## duration_numeric              -0.6799     0.8044  -0.845   0.3986
## sexFemale                     -2.1690     1.8583  -1.167   0.2439
## traveled_beforeAbroad before  -4.2962     1.8966  -2.265   0.0241 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.46 on 354 degrees of freedom
## Multiple R-squared:  0.01943,    Adjusted R-squared:  0.01112
## F-statistic: 2.338 on 3 and 354 DF,  p-value: 0.0733
```

Here we see no evidence of an effect of duration of stay, so taking the above results together, we conclude an absense of a relationship in this dataset. We'll output a table of this result.

```
texreg(duration_model, type = "html",  digits = 3, bold = .05, booktabs = TRUE, sideways = FALSE, use.pa
```

Finally we'll do a quick analysis of wealth class. We put 'extremely wealthy' into the 'upper class category' because there are only 2 'extremely wealthly'

```
wealth_model <- lm(stress ~ wealth_class_no_extreme + sex + traveled_before, data = refiltered)
summary(wealth_model)
```

```
##
## Call:
## lm(formula = stress ~ wealth_class_no_extreme + sex + traveled_before,
##     data = refiltered)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -31.791 -11.655  -1.806  10.577  53.473
##
## Coefficients:
```

```
##                                          Estimate Std. Error t value
## (Intercept)                               66.9941     5.0069  13.380
## wealth_class_no_extremeLower-middle class   0.1494     5.3007   0.028
## wealth_class_no_extremeMiddle class        -3.2302     4.9113  -0.658
## wealth_class_no_extremeUpper-middle class  -2.1637     5.0643  -0.427
## wealth_class_no_extremeUpper class         -5.7796     6.4289  -0.899
## sexFemale                                  -2.3035     1.8731  -1.230
## traveled_beforeAbroad before               -4.0495     1.9410  -2.086
##                                          Pr(>|t|)
## (Intercept)                                <2e-16 ***
## wealth_class_no_extremeLower-middle class   0.9775
## wealth_class_no_extremeMiddle class         0.5112
## wealth_class_no_extremeUpper-middle class   0.6695
## wealth_class_no_extremeUpper class          0.3693
## sexFemale                                   0.2196
## traveled_beforeAbroad before                0.0377 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 16.48 on 351 degrees of freedom
## Multiple R-squared:  0.02456,    Adjusted R-squared:  0.007884
## F-statistic: 1.473 on 6 and 351 DF,  p-value: 0.1865
```

```
Anova(wealth_model, type = "III")
```

```
## Anova Table (Type III tests)
##
## Response: stress
##                         Sum Sq  Df  F value  Pr(>F)
## (Intercept)              48649   1 179.0358 < 2e-16 ***
## wealth_class_no_extreme    695   4   0.6391 0.63490
## sex                        411   1   1.5124 0.21960
## traveled_before           1183   1   4.3529 0.03767 *
## Residuals                95376 351
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
ggplot(refiltered, aes(x=wealth_class_no_extreme, y=stress)) + geom_boxplot() + labs(x="Wealth class", y
```

	Stress
(Constant)	**66.994*****
	(5.007)
Lower-middle - Low	0.149
	(5.301)
Middle - Low	-3.230
	(4.911)
Upper-middle - Low	-2.164
	(5.064)
Upper - Low	-5.780
	(6.429)
Gender (Female)	-2.303
	(1.873)
Prior travel	$-$**4.050***
	(1.941)
R^2	0.025
Adj. R^2	0.008
Num. obs.	358

$^{***}p < 0.001$, $^{**}p < 0.01$, $^{*}p < 0.05$.
Results are as coefficient (standard error).

Table 12: No relationship between wealth class and stress was found

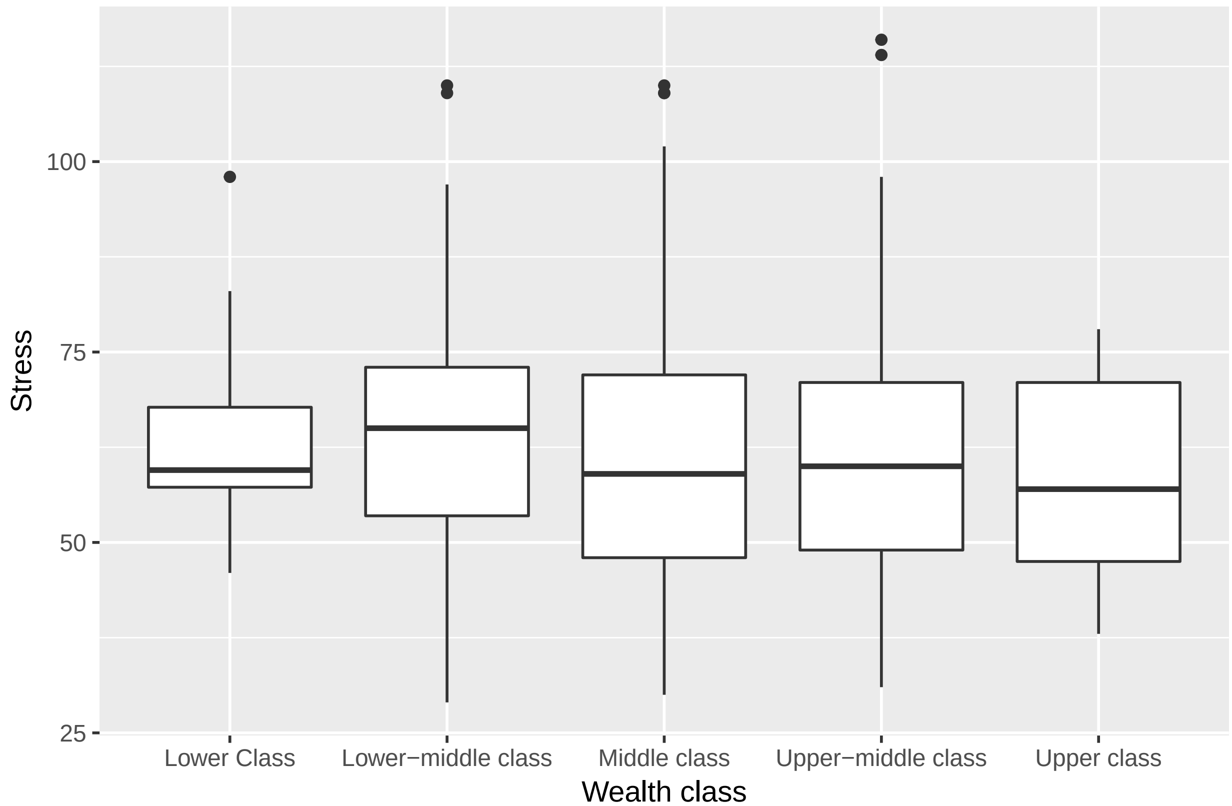


Figure 15: No relationship between reporeted wealth class and stress

```
texreg(wealth_model, type = "html",  digits = 3, bold = .05, booktabs = TRUE, sideways = FALSE, use.pac
```

The regression output indicated there was an effect of traveled before on stress - a result we've seen multiple times before - but it begs the question of whether those with higher incomes have traveled abroad before more in our sample. Let's do a chisq test to see if that's the case, and graph the result.

```
tbl <- table(refiltered$wealth_class_no_extreme, refiltered$traveled_before)
chisq.test(tbl)
```

```
## Warning in chisq.test(tbl): Chi-squared approximation may be incorrect
```

```
##
##  Pearson's Chi-squared test
##
## data:  tbl
## X-squared = 15.51, df = 4, p-value = 0.003753
```

```
per_abroad_before <-  refiltered %>% group_by(wealth_class_no_extreme) %>% summarise(per = mean(travele
ggplot(per_abroad_before, aes(x=wealth_class_no_extreme, y=per)) + geom_bar(stat="identity") + labs(x="
```

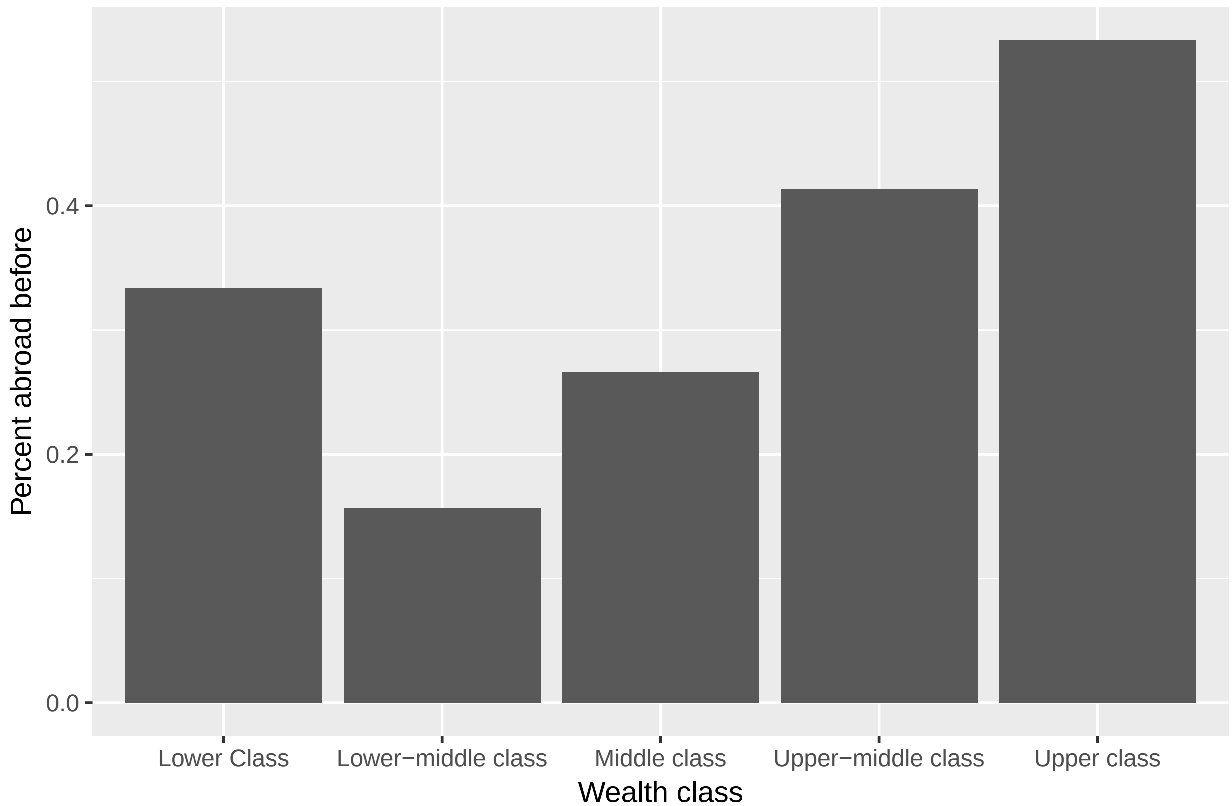


Figure 16: Wealth class is related to likelihood of prior abroad experiences

Mixed-models

Next we'll used a mixed model with data clustered by country where studying took place to look at heterogeneity in the effect of English confidence and test scores on stress.

```
stressMM <- mixed(stress ~ english_confidence +  sex + traveled_before + (1|country), data = noOtherCou
```

```
## Contrasts set to contr.sum for the following variables: sex, traveled_before, country
```

```
## Numerical variables NOT centered on 0 (i.e., interpretation of all main effects might be difficult i
```

```
## Fitting one lmer() model.

## Warning: contrasts dropped from factor country due to missing levels

## Warning: contrasts dropped from factor country due to missing levels

## [DONE]
## Calculating p-values.

## Warning: contrasts dropped from factor country due to missing levels

## Warning: contrasts dropped from factor country due to missing levels

## [DONE]
```

```
stressMMlmer <- lmer(stress ~ english_confidence + sex + traveled_before + (1|country), data = noOtherC
stressMMscores <- lmer(stress ~ overall_score_numeric_trimmed + sex + traveled_before + (1|country), da

summary(stressMM)
```

```
## Linear mixed model fit by REML ['merModLmerTest']
## Formula:
## stress ~ english_confidence + sex + traveled_before + (1 | country)
##    Data: noOtherCountry
##
## REML criterion at convergence: 2855.1
##
## Scaled residuals:
##      Min       1Q   Median       3Q      Max
## -1.91940 -0.72574 -0.08227  0.66558  3.13859
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  country  (Intercept)   3.714   1.927
##  Residual             257.473  16.046
## Number of obs: 341, groups:  country, 6
##
## Fixed effects:
##                     Estimate Std. Error t value
## (Intercept)          73.5568     4.0840  18.011
## english_confidence   -0.9259     0.2814  -3.290
## sex1                  0.6255     0.9417   0.664
## traveled_before1      2.0706     0.9573   2.163
##
## Correlation of Fixed Effects:
##             (Intr) engls_ sex1
## englsh_cnfd -0.947
## sex1         0.023  0.065
## travld_bfr1 -0.179  0.089  0.013
```

```
print(stressMM)
```

```
## Mixed Model Anova Table (Type 3 tests, KR-method)
##
## Model: stress ~ english_confidence + sex + traveled_before + (1 | country)
## Data: noOtherCountry
##               Effect        df        F p.value
```

```
## 1 english_confidence 1, 334.23 10.59 **    .001
## 2                sex 1, 333.51     0.43     .51
## 3    traveled_before 1, 336.48   4.59 *     .03
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '+' 0.1 ' ' 1
```

```
exactRLRT(m=stressMMlmer)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 1.2427, p-value = 0.0891
```

```
exactRLRT(m=stressMMscores)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 1.1725, p-value = 0.0948
```

A random intercept does not seem to be quite significant - what about a random slope model? or a random intercept + slope (no covariance so we can test 1 random effect at a time with `exactRLRT`)?

```
MM_RI <- stressMMlmer
MM_RI_RS_nocov <-  lmer(stress ~ english_confidence + sex + traveled_before + (1 + english_confidence||c
MM_RS <- lmer(stress ~ english_confidence + sex + traveled_before + (0 + english_confidence|country), da
MM_RS_scores <- lmer(stress ~ overall_score_numeric_trimmed + sex + traveled_before + (0 + overall_score
exactRLRT(m = MM_RS, mA = MM_RI_RS_nocov, m0 = MM_RI)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 0, p-value = 1
```

```
exactRLRT(m = MM_RS)
```

```
##
##  simulated finite sample distribution of RLRT.
##
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 0.33518, p-value = 0.2145
```

```
exactRLRT(m = MM_RS_scores)
```

```
##
##  simulated finite sample distribution of RLRT.
##
```

```
##  (p-value based on 10000 simulated values)
##
## data:
## RLRT = 0.82673, p-value = 0.1299
```

Neither are significant. Let's look at the random slope + intercept model anyway. See explanation above - negative correlation between random country intercepts and random slopes for English confidence, indicating that those countries higher stress also showed the most reduction in stress with higher test scores. This is also true for test scores.

```
MM_RI_RS <- lmer(stress ~ english_confidence + sex + traveled_before + (1 + english_confidence | country
summary(MM_RI_RS)
```

```
## Linear mixed model fit by REML t-tests use Satterthwaite approximations
##   to degrees of freedom [lmerMod]
## Formula:
## stress ~ english_confidence + sex + traveled_before + (1 + english_confidence |
##     country)
##    Data: noOtherCountry
##
## REML criterion at convergence: 2849.9
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.9325 -0.6758 -0.0881  0.6605  3.2042
##
## Random effects:
##  Groups   Name               Variance Std.Dev. Corr
##  country  (Intercept)         88.9723  9.4325
##           english_confidence   0.2656  0.5154  -1.00
##  Residual                    254.0125 15.9378
## Number of obs: 341, groups:  country, 6
##
## Fixed effects:
##                             Estimate Std. Error       df t value Pr(>|t|)
## (Intercept)                  74.2812     5.6488   6.3000  13.150 8.29e-06
## english_confidence           -0.7938     0.3513   7.5000  -2.259   0.0557
## sexFemale                    -1.0964     1.8683 333.0000  -0.587   0.5577
## traveled_beforeAbroad before  -4.3763     1.9021 333.8000  -2.301   0.0220
##
## (Intercept)                 ***
## english_confidence          .
## sexFemale
## traveled_beforeAbroad before *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) engls_ sexFml
## englsh_cnfd -0.944
## sexFemale   -0.192 -0.043
## trvld_bfrAb -0.031 -0.086  0.016
```

```
summary(lmer(stress ~ overall_score_numeric_trimmed + sex + traveled_before + (1 + overall_score_numeri
```

```
## Linear mixed model fit by REML t-tests use Satterthwaite approximations
```

```
##   to degrees of freedom [lmerMod]
## Formula: stress ~ overall_score_numeric_trimmed + sex + traveled_before +
##     (1 + overall_score_numeric_trimmed | country)
##    Data: noOtherCountry
##
## REML criterion at convergence: 2272.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.7813 -0.7304 -0.1726  0.6259  3.0950
##
## Random effects:
##  Groups   Name                          Variance Std.Dev. Corr
##  country  (Intercept)                   1262.84  35.537
##           overall_score_numeric_trimmed   23.39   4.837   -1.00
##  Residual                                282.78  16.816
## Number of obs: 269, groups:  country, 6
##
## Fixed effects:
##                               Estimate Std. Error       df t value
## (Intercept)                    69.1119    18.5681   3.4900   3.722
## overall_score_numeric_trimmed  -0.7719     2.5883   3.1600  -0.298
## sexFemale                      -1.6983     2.2058 258.3900  -0.770
## traveled_beforeAbroad before   -4.5380     2.2426 260.6700  -2.024
##                               Pr(>|t|)
## (Intercept)                     0.0259 *
## overall_score_numeric_trimmed   0.7841
## sexFemale                       0.4420
## traveled_beforeAbroad before    0.0440 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##             (Intr) ovr___ sexFml
## ovrll_scr__ -0.993
## sexFemale   -0.073 -0.008
## trvld_bfrAb  0.031 -0.074  0.011
```