

Feedback MTurk Study

Introduction

Load Data

```
d <- fread('Lungs_November+14,+2020_17.33.csv')
#head(d)

d_respondents_only <- d[ Status == "IP Address" , ]
#head(d_respondents_only)

#rename task phase questions
setnames(d_respondents_only,
  old = c('Q2', 'Q42'),
  new = c('Self_Reflect_Q1', 'Self_Reflect_Q2'))

setnames(d_respondents_only,
  old = c('Q69', 'Q89'),
  new = c('Control_Q1', 'Control_Q2'))

setnames(d_respondents_only,
  old = c('Q80', 'Q82', 'Q83', 'Q84', 'SC0', 'FL_6_D0'),
  new = c('Amazon_Turk_ID', 'Gender', 'Age_Range', 'Education_Level', 'Total_Score', 'Assignment'))

setnames(d_respondents_only,
  old = c('Q1', 'Q5', 'Q6', 'Q7', 'Q16', 'Q17', 'Q18', 'Q19', 'Q20', 'Q21',
    'Q8', 'Q9', 'Q10', 'Q11', 'Q22', 'Q23', 'Q24', 'Q25', 'Q26', 'Q27',
    'Q12', 'Q13', 'Q14', 'Q15', 'Q28', 'Q29', 'Q30', 'Q31', 'Q32', 'Q33'),
  new = c('Q1', 'Q2', 'Q3', 'Q4', 'Q5', 'Q6', 'Q7', 'Q8', 'Q9', 'Q10',
    'Q11', 'Q12', 'Q13', 'Q14', 'Q15', 'Q16', 'Q17', 'Q18', 'Q19', 'Q20',
    'Q21', 'Q22', 'Q23', 'Q24', 'Q25', 'Q26', 'Q27', 'Q28', 'Q29', 'Q30'))

d_respondents_only[, c("Q1_Score", "Q2_Score", "Q3_Score", "Q4_Score", "Q5_Score",
  "Q6_Score", "Q7_Score", "Q8_Score", "Q9_Score", "Q10_Score",
  "Q11_Score", "Q12_Score", "Q13_Score", "Q14_Score", "Q15_Score",
  "Q16_Score", "Q17_Score", "Q18_Score", "Q19_Score", "Q20_Score",
  "Q21_Score", "Q22_Score", "Q23_Score", "Q24_Score", "Q25_Score", "Q26_Score",
  "Q27_Score", "Q28_Score", "Q29_Score", "Q30_Score") :=
  list(ifelse(Q1 == "Normal", 1, 0),
    ifelse(Q2 == "Normal", 1, 0),
    ifelse(Q3 == "Pneumonia", 1, 0),
    ifelse(Q4 == "Pneumonia", 1, 0),
    ifelse(Q5 == "Normal", 1, 0),
    ifelse(Q6 == "Pneumonia", 1, 0),
```

```

        ifelse(Q7 == "Pneumonia", 1, 0),
        ifelse(Q8 == "Normal", 1, 0),
        ifelse(Q9 == "Pneumonia", 1, 0),
        ifelse(Q10 == "Normal", 1, 0),
        ifelse(Q11 == "Pneumonia", 1, 0),
        ifelse(Q12 == "Normal", 1, 0),
        ifelse(Q13 == "Pneumonia", 1, 0),
        ifelse(Q14 == "Pneumonia", 1, 0),
        ifelse(Q15 == "Normal", 1, 0),
        ifelse(Q16 == "Normal", 1, 0),
        ifelse(Q17 == "Pneumonia", 1, 0),
        ifelse(Q18 == "Normal", 1, 0),
        ifelse(Q19 == "Pneumonia", 1, 0),
        ifelse(Q20 == "Normal", 1, 0),
        ifelse(Q21 == "Normal", 1, 0),
        ifelse(Q22 == "Normal", 1, 0),
        ifelse(Q23 == "Pneumonia", 1, 0),
        ifelse(Q24 == "Normal", 1, 0),
        ifelse(Q25 == "Pneumonia", 1, 0),
        ifelse(Q26 == "Pneumonia", 1, 0),
        ifelse(Q27 == "Pneumonia", 1, 0),
        ifelse(Q28 == "Pneumonia", 1, 0),
        ifelse(Q29 == "Normal", 1, 0),
        ifelse(Q30 == "Normal", 1, 0))]

d_respondents_only[ , Assignment_Group := ifelse(Assignment == "FL_17", "Control",
        ifelse(Assignment == "FL_14", "Self-Reflect",
        ifelse(Assignment == "FL_15", "Medical Feedback",
        ifelse(Assignment == "FL_16", "Positive Images", "Negative Images"))

d_respondents_only[ , c("TaskPhase1_Score", "TaskPhase2_Score", "TaskPhase3_Score") :=
        list(sum(Q1_Score, Q2_Score, Q3_Score, Q4_Score, Q5_Score, Q6_Score, Q7_Score, Q8_Score,
        sum(Q11_Score, Q12_Score, Q13_Score, Q14_Score, Q15_Score, Q16_Score, Q17_Score, Q18_Score,
        sum(Q21_Score, Q22_Score, Q23_Score, Q24_Score, Q25_Score, Q26_Score, Q27_Score, Q28_Score,
        by = Amazon_Turk_ID]

#head(d_respondents_only)

# ?register_google
# register_google(key = "AIzaSyCTk2a5vIEqcvgz9KmQmItoNF7J8_hiMMk")

#uses Google API to obtain location data based on longitude and latitude....dont use unless necessary f
# d_respondents_only[ , c("houseNumber", "street", "city", "county", "state", "zip", "country") := revgeog
#
# #head(d_respondents_only)
#
#
# fwrite(d_respondents_only, file='datatable_clean_survey_responses.dta')

d_respondents <- fread('datatable_clean_survey_responses.dta')
#head(d_respondents)

#skip

```

```

# ?register_google
# register_google(key = "AIzaSyCTk2a5vIEqcvgz9KmQmItoNF7J8_hiMMk")
# ggmap_show_api_key()
#
#
# revgeocode(c(df$lon[1], df$lat[1]))
#
# d_respondents_only[ Q80 == "A1AC47WJLW4G7", revgeocode(c(as.numeric(LocationLongitude)[1], as.numeri
# ?revgeocode

#remove duplicate Amazon Turk IDs
nrow(d_respondents) #381 rows

## [1] 381

d_respondents <- d_respondents[ !duplicated(d_respondents$Amazon_Turk_ID) , ] #378 rows

```

EDA

```

#some EDA

#d_respondents[, table(state, country)]

table(d_respondents$state, d_respondents$country) %>%
  as.data.frame() %>%
  arrange(desc(Freq))

```

##	Var1	Var2	Freq
## 1	Tamil Nadu	India	120
## 2	California	United States	82
## 3	New York	United States	22
## 4	Kansas	United States	21
## 5	Texas	United States	16
## 6	Florida	United States	9
## 7	Massachusetts	United States	7
## 8	Michigan	United States	7
## 9	Missouri	United States	6
## 10	Connecticut	United States	5
## 11	Georgia	United States	5
## 12	Indiana	United States	5
## 13	New Jersey	United States	5
## 14	Virginia	United States	5
## 15	Illinois	United States	4
## 16	North Carolina	United States	4
## 17	Kerala	India	3
## 18	Maharashtra	India	3
## 19	Colorado	United States	3
## 20	Kentucky	United States	3
## 21	Maryland	United States	3
## 22	Oregon	United States	3
## 23	Ontario	Canada	2
## 24	Alabama	United States	2
## 25	Idaho	United States	2

## 26	Minnesota	United States	2
## 27	Mississippi	United States	2
## 28	Nevada	United States	2
## 29	Ohio	United States	2
## 30	Pennsylvania	United States	2
## 31	Washington	United States	2
## 32	Qarku i Tiranës	Albania	1
## 33	Khulna Division	Bangladesh	1
## 34	Bahia	Brazil	1
## 35	Atacama	Chile	1
## 36	Provence-Alpes-Côte d'Azur	France	1
## 37	Departamento de Olancho	Honduras	1
## 38	Andhra Pradesh	India	1
## 39	Karnataka	India	1
## 40	Sardegna	Italy	1
## 41	England	United Kingdom	1
## 42	Arizona	United States	1
## 43	Iowa	United States	1
## 44	Louisiana	United States	1
## 45	Maine	United States	1
## 46	Nebraska	United States	1
## 47	Oklahoma	United States	1
## 48	South Carolina	United States	1
## 49	South Dakota	United States	1
## 50	Tennessee	United States	1
## 51	Alabama	Albania	0
## 52	Andhra Pradesh	Albania	0
## 53	Arizona	Albania	0
## 54	Atacama	Albania	0
## 55	Bahia	Albania	0
## 56	California	Albania	0
## 57	Colorado	Albania	0
## 58	Connecticut	Albania	0
## 59	Departamento de Olancho	Albania	0
## 60	England	Albania	0
## 61	Florida	Albania	0
## 62	Georgia	Albania	0
## 63	Idaho	Albania	0
## 64	Illinois	Albania	0
## 65	Indiana	Albania	0
## 66	Iowa	Albania	0
## 67	Kansas	Albania	0
## 68	Karnataka	Albania	0
## 69	Kentucky	Albania	0
## 70	Kerala	Albania	0
## 71	Khulna Division	Albania	0
## 72	Louisiana	Albania	0
## 73	Maharashtra	Albania	0
## 74	Maine	Albania	0
## 75	Maryland	Albania	0
## 76	Massachusetts	Albania	0
## 77	Michigan	Albania	0
## 78	Minnesota	Albania	0
## 79	Mississippi	Albania	0

## 80	Missouri	Albania	0
## 81	Nebraska	Albania	0
## 82	Nevada	Albania	0
## 83	New Jersey	Albania	0
## 84	New York	Albania	0
## 85	North Carolina	Albania	0
## 86	Ohio	Albania	0
## 87	Oklahoma	Albania	0
## 88	Ontario	Albania	0
## 89	Oregon	Albania	0
## 90	Pennsylvania	Albania	0
## 91	Provence-Alpes-Côte d'Azur	Albania	0
## 92	Sardegna	Albania	0
## 93	South Carolina	Albania	0
## 94	South Dakota	Albania	0
## 95	Tamil Nadu	Albania	0
## 96	Tennessee	Albania	0
## 97	Texas	Albania	0
## 98	Virginia	Albania	0
## 99	Washington	Albania	0
## 100	Alabama	Bangladesh	0
## 101	Andhra Pradesh	Bangladesh	0
## 102	Arizona	Bangladesh	0
## 103	Atacama	Bangladesh	0
## 104	Bahia	Bangladesh	0
## 105	California	Bangladesh	0
## 106	Colorado	Bangladesh	0
## 107	Connecticut	Bangladesh	0
## 108	Departamento de Olancho	Bangladesh	0
## 109	England	Bangladesh	0
## 110	Florida	Bangladesh	0
## 111	Georgia	Bangladesh	0
## 112	Idaho	Bangladesh	0
## 113	Illinois	Bangladesh	0
## 114	Indiana	Bangladesh	0
## 115	Iowa	Bangladesh	0
## 116	Kansas	Bangladesh	0
## 117	Karnataka	Bangladesh	0
## 118	Kentucky	Bangladesh	0
## 119	Kerala	Bangladesh	0
## 120	Louisiana	Bangladesh	0
## 121	Maharashtra	Bangladesh	0
## 122	Maine	Bangladesh	0
## 123	Maryland	Bangladesh	0
## 124	Massachusetts	Bangladesh	0
## 125	Michigan	Bangladesh	0
## 126	Minnesota	Bangladesh	0
## 127	Mississippi	Bangladesh	0
## 128	Missouri	Bangladesh	0
## 129	Nebraska	Bangladesh	0
## 130	Nevada	Bangladesh	0
## 131	New Jersey	Bangladesh	0
## 132	New York	Bangladesh	0
## 133	North Carolina	Bangladesh	0

## 134	Ohio	Bangladesh	0
## 135	Oklahoma	Bangladesh	0
## 136	Ontario	Bangladesh	0
## 137	Oregon	Bangladesh	0
## 138	Pennsylvania	Bangladesh	0
## 139	Provence-Alpes-Côte d'Azur	Bangladesh	0
## 140	Qarku i Tiranës	Bangladesh	0
## 141	Sardegna	Bangladesh	0
## 142	South Carolina	Bangladesh	0
## 143	South Dakota	Bangladesh	0
## 144	Tamil Nadu	Bangladesh	0
## 145	Tennessee	Bangladesh	0
## 146	Texas	Bangladesh	0
## 147	Virginia	Bangladesh	0
## 148	Washington	Bangladesh	0
## 149	Alabama	Brazil	0
## 150	Andhra Pradesh	Brazil	0
## 151	Arizona	Brazil	0
## 152	Atacama	Brazil	0
## 153	California	Brazil	0
## 154	Colorado	Brazil	0
## 155	Connecticut	Brazil	0
## 156	Departamento de Olancho	Brazil	0
## 157	England	Brazil	0
## 158	Florida	Brazil	0
## 159	Georgia	Brazil	0
## 160	Idaho	Brazil	0
## 161	Illinois	Brazil	0
## 162	Indiana	Brazil	0
## 163	Iowa	Brazil	0
## 164	Kansas	Brazil	0
## 165	Karnataka	Brazil	0
## 166	Kentucky	Brazil	0
## 167	Kerala	Brazil	0
## 168	Khulna Division	Brazil	0
## 169	Louisiana	Brazil	0
## 170	Maharashtra	Brazil	0
## 171	Maine	Brazil	0
## 172	Maryland	Brazil	0
## 173	Massachusetts	Brazil	0
## 174	Michigan	Brazil	0
## 175	Minnesota	Brazil	0
## 176	Mississippi	Brazil	0
## 177	Missouri	Brazil	0
## 178	Nebraska	Brazil	0
## 179	Nevada	Brazil	0
## 180	New Jersey	Brazil	0
## 181	New York	Brazil	0
## 182	North Carolina	Brazil	0
## 183	Ohio	Brazil	0
## 184	Oklahoma	Brazil	0
## 185	Ontario	Brazil	0
## 186	Oregon	Brazil	0
## 187	Pennsylvania	Brazil	0

## 188	Provence-Alpes-Côte d'Azur	Brazil	0
## 189	Qarku i Tiranës	Brazil	0
## 190	Sardegna	Brazil	0
## 191	South Carolina	Brazil	0
## 192	South Dakota	Brazil	0
## 193	Tamil Nadu	Brazil	0
## 194	Tennessee	Brazil	0
## 195	Texas	Brazil	0
## 196	Virginia	Brazil	0
## 197	Washington	Brazil	0
## 198	Alabama	Canada	0
## 199	Andhra Pradesh	Canada	0
## 200	Arizona	Canada	0
## 201	Atacama	Canada	0
## 202	Bahia	Canada	0
## 203	California	Canada	0
## 204	Colorado	Canada	0
## 205	Connecticut	Canada	0
## 206	Departamento de Olancho	Canada	0
## 207	England	Canada	0
## 208	Florida	Canada	0
## 209	Georgia	Canada	0
## 210	Idaho	Canada	0
## 211	Illinois	Canada	0
## 212	Indiana	Canada	0
## 213	Iowa	Canada	0
## 214	Kansas	Canada	0
## 215	Karnataka	Canada	0
## 216	Kentucky	Canada	0
## 217	Kerala	Canada	0
## 218	Khulna Division	Canada	0
## 219	Louisiana	Canada	0
## 220	Maharashtra	Canada	0
## 221	Maine	Canada	0
## 222	Maryland	Canada	0
## 223	Massachusetts	Canada	0
## 224	Michigan	Canada	0
## 225	Minnesota	Canada	0
## 226	Mississippi	Canada	0
## 227	Missouri	Canada	0
## 228	Nebraska	Canada	0
## 229	Nevada	Canada	0
## 230	New Jersey	Canada	0
## 231	New York	Canada	0
## 232	North Carolina	Canada	0
## 233	Ohio	Canada	0
## 234	Oklahoma	Canada	0
## 235	Oregon	Canada	0
## 236	Pennsylvania	Canada	0
## 237	Provence-Alpes-Côte d'Azur	Canada	0
## 238	Qarku i Tiranës	Canada	0
## 239	Sardegna	Canada	0
## 240	South Carolina	Canada	0
## 241	South Dakota	Canada	0

## 242	Tamil Nadu	Canada	0
## 243	Tennessee	Canada	0
## 244	Texas	Canada	0
## 245	Virginia	Canada	0
## 246	Washington	Canada	0
## 247	Alabama	Chile	0
## 248	Andhra Pradesh	Chile	0
## 249	Arizona	Chile	0
## 250	Bahia	Chile	0
## 251	California	Chile	0
## 252	Colorado	Chile	0
## 253	Connecticut	Chile	0
## 254	Departamento de Olancho	Chile	0
## 255	England	Chile	0
## 256	Florida	Chile	0
## 257	Georgia	Chile	0
## 258	Idaho	Chile	0
## 259	Illinois	Chile	0
## 260	Indiana	Chile	0
## 261	Iowa	Chile	0
## 262	Kansas	Chile	0
## 263	Karnataka	Chile	0
## 264	Kentucky	Chile	0
## 265	Kerala	Chile	0
## 266	Khulna Division	Chile	0
## 267	Louisiana	Chile	0
## 268	Maharashtra	Chile	0
## 269	Maine	Chile	0
## 270	Maryland	Chile	0
## 271	Massachusetts	Chile	0
## 272	Michigan	Chile	0
## 273	Minnesota	Chile	0
## 274	Mississippi	Chile	0
## 275	Missouri	Chile	0
## 276	Nebraska	Chile	0
## 277	Nevada	Chile	0
## 278	New Jersey	Chile	0
## 279	New York	Chile	0
## 280	North Carolina	Chile	0
## 281	Ohio	Chile	0
## 282	Oklahoma	Chile	0
## 283	Ontario	Chile	0
## 284	Oregon	Chile	0
## 285	Pennsylvania	Chile	0
## 286	Provence-Alpes-Côte d'Azur	Chile	0
## 287	Qarku i Tiranës	Chile	0
## 288	Sardegna	Chile	0
## 289	South Carolina	Chile	0
## 290	South Dakota	Chile	0
## 291	Tamil Nadu	Chile	0
## 292	Tennessee	Chile	0
## 293	Texas	Chile	0
## 294	Virginia	Chile	0
## 295	Washington	Chile	0

## 296	Alabama	France	0
## 297	Andhra Pradesh	France	0
## 298	Arizona	France	0
## 299	Atacama	France	0
## 300	Bahia	France	0
## 301	California	France	0
## 302	Colorado	France	0
## 303	Connecticut	France	0
## 304	Departamento de Olancho	France	0
## 305	England	France	0
## 306	Florida	France	0
## 307	Georgia	France	0
## 308	Idaho	France	0
## 309	Illinois	France	0
## 310	Indiana	France	0
## 311	Iowa	France	0
## 312	Kansas	France	0
## 313	Karnataka	France	0
## 314	Kentucky	France	0
## 315	Kerala	France	0
## 316	Khulna Division	France	0
## 317	Louisiana	France	0
## 318	Maharashtra	France	0
## 319	Maine	France	0
## 320	Maryland	France	0
## 321	Massachusetts	France	0
## 322	Michigan	France	0
## 323	Minnesota	France	0
## 324	Mississippi	France	0
## 325	Missouri	France	0
## 326	Nebraska	France	0
## 327	Nevada	France	0
## 328	New Jersey	France	0
## 329	New York	France	0
## 330	North Carolina	France	0
## 331	Ohio	France	0
## 332	Oklahoma	France	0
## 333	Ontario	France	0
## 334	Oregon	France	0
## 335	Pennsylvania	France	0
## 336	Qarku i Tiranës	France	0
## 337	Sardegna	France	0
## 338	South Carolina	France	0
## 339	South Dakota	France	0
## 340	Tamil Nadu	France	0
## 341	Tennessee	France	0
## 342	Texas	France	0
## 343	Virginia	France	0
## 344	Washington	France	0
## 345	Alabama	Honduras	0
## 346	Andhra Pradesh	Honduras	0
## 347	Arizona	Honduras	0
## 348	Atacama	Honduras	0
## 349	Bahia	Honduras	0

## 350	California	Honduras	0
## 351	Colorado	Honduras	0
## 352	Connecticut	Honduras	0
## 353	England	Honduras	0
## 354	Florida	Honduras	0
## 355	Georgia	Honduras	0
## 356	Idaho	Honduras	0
## 357	Illinois	Honduras	0
## 358	Indiana	Honduras	0
## 359	Iowa	Honduras	0
## 360	Kansas	Honduras	0
## 361	Karnataka	Honduras	0
## 362	Kentucky	Honduras	0
## 363	Kerala	Honduras	0
## 364	Khulna Division	Honduras	0
## 365	Louisiana	Honduras	0
## 366	Maharashtra	Honduras	0
## 367	Maine	Honduras	0
## 368	Maryland	Honduras	0
## 369	Massachusetts	Honduras	0
## 370	Michigan	Honduras	0
## 371	Minnesota	Honduras	0
## 372	Mississippi	Honduras	0
## 373	Missouri	Honduras	0
## 374	Nebraska	Honduras	0
## 375	Nevada	Honduras	0
## 376	New Jersey	Honduras	0
## 377	New York	Honduras	0
## 378	North Carolina	Honduras	0
## 379	Ohio	Honduras	0
## 380	Oklahoma	Honduras	0
## 381	Ontario	Honduras	0
## 382	Oregon	Honduras	0
## 383	Pennsylvania	Honduras	0
## 384	Provence-Alpes-Côte d'Azur	Honduras	0
## 385	Qarku i Tiranës	Honduras	0
## 386	Sardegna	Honduras	0
## 387	South Carolina	Honduras	0
## 388	South Dakota	Honduras	0
## 389	Tamil Nadu	Honduras	0
## 390	Tennessee	Honduras	0
## 391	Texas	Honduras	0
## 392	Virginia	Honduras	0
## 393	Washington	Honduras	0
## 394	Alabama	India	0
## 395	Arizona	India	0
## 396	Atacama	India	0
## 397	Bahia	India	0
## 398	California	India	0
## 399	Colorado	India	0
## 400	Connecticut	India	0
## 401	Departamento de Olancho	India	0
## 402	England	India	0
## 403	Florida	India	0

## 404	Georgia	India	0
## 405	Idaho	India	0
## 406	Illinois	India	0
## 407	Indiana	India	0
## 408	Iowa	India	0
## 409	Kansas	India	0
## 410	Kentucky	India	0
## 411	Khulna Division	India	0
## 412	Louisiana	India	0
## 413	Maine	India	0
## 414	Maryland	India	0
## 415	Massachusetts	India	0
## 416	Michigan	India	0
## 417	Minnesota	India	0
## 418	Mississippi	India	0
## 419	Missouri	India	0
## 420	Nebraska	India	0
## 421	Nevada	India	0
## 422	New Jersey	India	0
## 423	New York	India	0
## 424	North Carolina	India	0
## 425	Ohio	India	0
## 426	Oklahoma	India	0
## 427	Ontario	India	0
## 428	Oregon	India	0
## 429	Pennsylvania	India	0
## 430	Provence-Alpes-Côte d'Azur	India	0
## 431	Qarku i Tiranës	India	0
## 432	Sardegna	India	0
## 433	South Carolina	India	0
## 434	South Dakota	India	0
## 435	Tennessee	India	0
## 436	Texas	India	0
## 437	Virginia	India	0
## 438	Washington	India	0
## 439	Alabama	Italy	0
## 440	Andhra Pradesh	Italy	0
## 441	Arizona	Italy	0
## 442	Atacama	Italy	0
## 443	Bahia	Italy	0
## 444	California	Italy	0
## 445	Colorado	Italy	0
## 446	Connecticut	Italy	0
## 447	Departamento de Olancho	Italy	0
## 448	England	Italy	0
## 449	Florida	Italy	0
## 450	Georgia	Italy	0
## 451	Idaho	Italy	0
## 452	Illinois	Italy	0
## 453	Indiana	Italy	0
## 454	Iowa	Italy	0
## 455	Kansas	Italy	0
## 456	Karnataka	Italy	0
## 457	Kentucky	Italy	0

## 458	Kerala	Italy	0
## 459	Khulna Division	Italy	0
## 460	Louisiana	Italy	0
## 461	Maharashtra	Italy	0
## 462	Maine	Italy	0
## 463	Maryland	Italy	0
## 464	Massachusetts	Italy	0
## 465	Michigan	Italy	0
## 466	Minnesota	Italy	0
## 467	Mississippi	Italy	0
## 468	Missouri	Italy	0
## 469	Nebraska	Italy	0
## 470	Nevada	Italy	0
## 471	New Jersey	Italy	0
## 472	New York	Italy	0
## 473	North Carolina	Italy	0
## 474	Ohio	Italy	0
## 475	Oklahoma	Italy	0
## 476	Ontario	Italy	0
## 477	Oregon	Italy	0
## 478	Pennsylvania	Italy	0
## 479	Provence-Alpes-Côte d'Azur	Italy	0
## 480	Qarku i Tiranës	Italy	0
## 481	South Carolina	Italy	0
## 482	South Dakota	Italy	0
## 483	Tamil Nadu	Italy	0
## 484	Tennessee	Italy	0
## 485	Texas	Italy	0
## 486	Virginia	Italy	0
## 487	Washington	Italy	0
## 488	Alabama	United Kingdom	0
## 489	Andhra Pradesh	United Kingdom	0
## 490	Arizona	United Kingdom	0
## 491	Atacama	United Kingdom	0
## 492	Bahia	United Kingdom	0
## 493	California	United Kingdom	0
## 494	Colorado	United Kingdom	0
## 495	Connecticut	United Kingdom	0
## 496	Departamento de Olancho	United Kingdom	0
## 497	Florida	United Kingdom	0
## 498	Georgia	United Kingdom	0
## 499	Idaho	United Kingdom	0
## 500	Illinois	United Kingdom	0
## 501	Indiana	United Kingdom	0
## 502	Iowa	United Kingdom	0
## 503	Kansas	United Kingdom	0
## 504	Karnataka	United Kingdom	0
## 505	Kentucky	United Kingdom	0
## 506	Kerala	United Kingdom	0
## 507	Khulna Division	United Kingdom	0
## 508	Louisiana	United Kingdom	0
## 509	Maharashtra	United Kingdom	0
## 510	Maine	United Kingdom	0
## 511	Maryland	United Kingdom	0

```
## 512      Massachusetts United Kingdom 0
## 513      Michigan United Kingdom 0
## 514      Minnesota United Kingdom 0
## 515      Mississippi United Kingdom 0
## 516      Missouri United Kingdom 0
## 517      Nebraska United Kingdom 0
## 518      Nevada United Kingdom 0
## 519      New Jersey United Kingdom 0
## 520      New York United Kingdom 0
## 521      North Carolina United Kingdom 0
## 522      Ohio United Kingdom 0
## 523      Oklahoma United Kingdom 0
## 524      Ontario United Kingdom 0
## 525      Oregon United Kingdom 0
## 526      Pennsylvania United Kingdom 0
## 527 Provence-Alpes-Côte d'Azur United Kingdom 0
## 528      Qarku i Tiranës United Kingdom 0
## 529      Sardegna United Kingdom 0
## 530      South Carolina United Kingdom 0
## 531      South Dakota United Kingdom 0
## 532      Tamil Nadu United Kingdom 0
## 533      Tennessee United Kingdom 0
## 534      Texas United Kingdom 0
## 535      Virginia United Kingdom 0
## 536      Washington United Kingdom 0
## 537      Andhra Pradesh United States 0
## 538      Atacama United States 0
## 539      Bahia United States 0
## 540      Departamento de Olancho United States 0
## 541      England United States 0
## 542      Karnataka United States 0
## 543      Kerala United States 0
## 544      Khulna Division United States 0
## 545      Maharashtra United States 0
## 546      Ontario United States 0
## 547 Provence-Alpes-Côte d'Azur United States 0
## 548      Qarku i Tiranës United States 0
## 549      Sardegna United States 0
## 550      Tamil Nadu United States 0
```

```
table(d_respondents$country) %>%
  as.data.frame() %>%
  arrange(desc(Freq))
```

```
##      Var1 Freq
## 1 United States 240
## 2      India 128
## 3      Canada   2
## 4      Albania   1
## 5  Bangladesh   1
## 6      Brazil   1
## 7      Chile   1
## 8      France   1
## 9      Honduras 1
## 10     Italy    1
```

```
## 11 United Kingdom      1
```

```
table(d_respondents$Total_Score) %>%  
  as.data.frame() %>%  
  arrange(desc(Var1))
```

```
##      Var1 Freq  
## 1      27     1  
## 2      26     1  
## 3      25     4  
## 4      24    13  
## 5      23    15  
## 6      22    16  
## 7      21    23  
## 8      20    27  
## 9      19    21  
## 10     18    32  
## 11     17    40  
## 12     16    40  
## 13     15    46  
## 14     14    31  
## 15     13    19  
## 16     12    19  
## 17     11    13  
## 18     10    13  
## 19      9     3  
## 20      8     1
```

```
d_respondents %>%  
  group_by(Assignment_Group) %>%  
  summarise(mean = mean(Total_Score),  
            count = n(),  
            time_duration = mean(`Duration (in seconds)`))
```

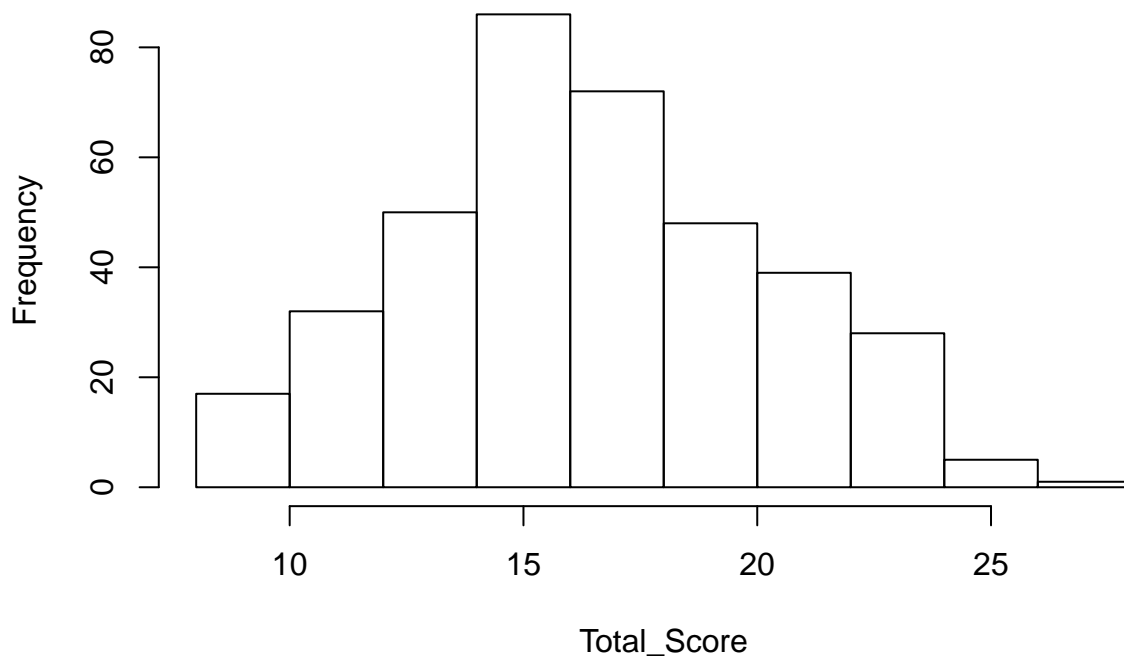
```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 5 x 4  
##   Assignment_Group mean count time_duration  
##   <chr>          <dbl> <int>      <dbl>  
## 1 Control         16.4    76        623.  
## 2 Medical Feedback 17.6    76        646.  
## 3 Negative Images  16.5    77        772.  
## 4 Positive Images  17.0    76        514.  
## 5 Self-Reflect    17.0    73        609.
```

```
#d_respondents[ , .(count = .N, avg = mean(Total_Score)), by=Assignment_Group] #same thing
```

```
d_respondents[ , hist(Total_Score)]
```

Histogram of Total_Score



```
## $breaks
## [1]  8 10 12 14 16 18 20 22 24 26 28
##
## $counts
## [1] 17 32 50 86 72 48 39 28  5  1
##
## $density
## [1] 0.022486772 0.042328042 0.066137566 0.113756614 0.095238095 0.063492063
## [7] 0.051587302 0.037037037 0.006613757 0.001322751
##
## $mids
## [1]  9 11 13 15 17 19 21 23 25 27
##
## $xname
## [1] "Total_Score"
##
## $equidist
## [1] TRUE
##
## attr(,"class")
## [1] "histogram"
```

```
tapply(d_respondents$Total_Score, d_respondents$Assignment_Group, summary)
```

```
## $Control
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##   8.00  14.00   16.00   16.41  19.00   24.00
##
## $`Medical Feedback`
##   Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
```

```
##    10.00    15.00    17.00    17.61    20.00    24.00
##
## $`Negative Images`
##    Min. 1st Qu.  Median      Mean 3rd Qu.    Max.
##    9.00   14.00   16.00   16.45   19.00   25.00
##
## $`Positive Images`
##    Min. 1st Qu.  Median      Mean 3rd Qu.    Max.
##    9.00   14.00   17.00   17.04   20.00   27.00
##
## $`Self-Reflect`
##    Min. 1st Qu.  Median      Mean 3rd Qu.    Max.
##    9.00   14.00   17.00   16.99   20.00   25.00
```

```
tapply(d_respondents$Total_Score, d_respondents$Assignment_Group, sd)
```

```
##           Control Medical Feedback  Negative Images  Positive Images
##           3.666707           3.417909           3.881622           3.930872
##           Self-Reflect
##           3.970352
```

```
d_respondents[, sd(Total_Score)]
```

```
## [1] 3.783594
```

Randomization Check

```
#http://www.sthda.com/english/wiki/chi-square-goodness-of-fit-test-in-r
```

```
respondent_counts <- d_respondents[, .(N), keyby=Assignment_Group][,2]
```

```
respondent_counts_chisq_test <- chisq.test(respondent_counts, p=c(1/5, 1/5, 1/5, 1/5, 1/5))
```

```
respondent_counts_chisq_test
```

```
##
## Chi-squared test for given probabilities
##
## data:  respondent_counts
## X-squared = 0.12169, df = 4, p-value = 0.9982
```

```
#p-value = 0.9982, which is greater than significance level of 0.05.
```

```
#We can conclude that the observed proportions are not significantly different from the expected proportions
```

Covariate Balance Check

```
d_respondents %>%
  group_by(Assignment_Group) %>%
  summarise(num_respondents = n(),
            pre_treatment_avg = mean(TaskPhase1_Score),
            taskphase2_avg = mean(TaskPhase2_Score),
            taskphase3_avg = mean(TaskPhase3_Score))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 5 x 5
```



```
## Assignment_Group num_respondents pre_treatment_avg taskphase2_avg
## <chr> <int> <dbl> <dbl>
## 1 Control 76 5.95 4.55
## 2 Medical Feedback 76 6.41 5.32
## 3 Negative Images 77 5.71 4.91
## 4 Positive Images 76 6.16 5.13
## 5 Self-Reflect 73 5.90 5.16
## # ... with 1 more variable: taskphase3_avg <dbl>
```

```
#check balance between age-range, education, age
```

```
d_respondents[ , table(Assignment_Group, Gender)]
```

```
## Gender
## Assignment_Group Female Male
## Control 27 49
## Medical Feedback 31 45
## Negative Images 31 46
## Positive Images 31 45
## Self-Reflect 30 43
```

```
d_respondents[ , table(Assignment_Group, Age_Range)]
```

```
## Age_Range
## Assignment_Group 18-24 25-34 35-44 45-54 55-64 Above 65
## Control 5 39 14 8 9 1
## Medical Feedback 5 40 16 7 6 2
## Negative Images 4 41 16 9 7 0
## Positive Images 4 34 21 6 11 0
## Self-Reflect 4 37 10 12 9 1
```

```
d_respondents[ , table(Assignment_Group, Education_Level)]
```

```
## Education_Level
## Assignment_Group Associate's degree Bachelor's degree High school
## Control 3 47 1
## Medical Feedback 0 59 1
## Negative Images 2 54 3
## Positive Images 4 50 0
## Self-Reflect 4 49 7
```

```
## Education_Level
## Assignment_Group Master's degree and above Some high school Trade school
## Control 22 2 1
## Medical Feedback 15 0 1
## Negative Images 14 1 3
## Positive Images 20 0 2
## Self-Reflect 12 0 1
```

```
#let's consider adding age bins and education bins
```

```
d_respondents[ Age_Range == "18-24", age_bin := 1]
d_respondents[ Age_Range == "25-34", age_bin := 2]
d_respondents[ Age_Range == "35-44", age_bin := 3]
d_respondents[ Age_Range == "45-54", age_bin := 4]
d_respondents[ Age_Range == "55-64", age_bin := 5]
d_respondents[ Age_Range == "Above 65", age_bin := 6]
```

```

d_respondents[ Education_Level == "Associate's degree", edu_bin := 1]
d_respondents[ Education_Level == "Bachelor's degree", edu_bin := 2]
d_respondents[ Education_Level == "High school", edu_bin := 3]
d_respondents[ Education_Level == "Master's degree and above", edu_bin := 4]
d_respondents[ Education_Level == "Some high school", edu_bin := 5]
d_respondents[ Education_Level == "Trade school", edu_bin := 6]

d_respondents[ Assignment_Group == "Control", assign_bin := 1]
d_respondents[ Assignment_Group == "Medical Feedback", assign_bin := 2]
d_respondents[ Assignment_Group == "Negative Images", assign_bin := 3]
d_respondents[ Assignment_Group == "Positive Images", assign_bin := 4]
d_respondents[ Assignment_Group == "Self-Reflect", assign_bin := 5]

#head(d_respondents)

d_respondents[ , Treatment_Dummy := ifelse(Assignment_Group != "Control", 1, 0)]
#head(d_respondents)

d_respondents[ Treatment_Dummy == 1, mean(Total_Score)] - d_respondents[ Treatment_Dummy == 0, mean(Tot

## [1] 0.6119728
sd(d_respondents$Total_Score)

## [1] 3.783594
d_respondents[ , lm(Total_Score ~ Education_Level)]

##
## Call:
## lm(formula = Total_Score ~ Education_Level)
##
## Coefficients:
##                (Intercept)
##                19.15385
##      Education_LevelBachelor's degree
##                -2.48975
##      Education_LevelHigh school
##                0.09615
## Education_LevelMaster's degree and above
##                -2.34662
##      Education_LevelSome high school
##                -3.15385
##      Education_LevelTrade school
##                -0.65385

d_respondents[ , ivreg(Total_Score ~ Education_Level | Assignment_Group)]

## Warning in ivreg.fit(X, Y, Z, weights, offset, ...): more regressors than
## instruments

##
## Call:
## ivreg(formula = Total_Score ~ Education_Level | Assignment_Group)
##
## Coefficients:
##                (Intercept)

```

```
##                                5.097
##      Education_LevelBachelor's degree
##                                11.909
##      Education_LevelHigh school
##                                13.868
## Education_LevelMaster's degree and above
##                                15.609
##      Education_LevelSome high school
##                                -28.680
##      Education_LevelTrade school
##                                NA

power.t.test( delta = 1.2, sd=3.78, sig.level = 0.05, power=0.8)

##
##      Two-sample t test power calculation
##
##      n = 156.7272
##      delta = 1.2
##      sd = 3.78
##      sig.level = 0.05
##      power = 0.8
##      alternative = two.sided
##
## NOTE: n is number in *each* group
```

Analysis

Helper Functions

```
get_robust_se <- function(model){
  # Get robust SE for use in stargazer
  vcov <- vcovHC(model)
  return(sqrt(diag(vcov)))
}
```

Total Score Analysis

```
#does treatment have an effect on total score?
mod1 <- d_respondents[, lm(Total_Score ~ Treatment_Dummy)]

#does treatment and pretreatment score have an effect on total score?
mod2 <- d_respondents[, lm(Total_Score ~ Treatment_Dummy + TaskPhase1_Score)]

stargazer(mod1,
           mod2,
           se = list(get_robust_se(mod1),get_robust_se(mod2)),
           type='text')

##
## =====
##                                Dependent variable:
##      -----
##                                Total_Score
```

```
##                                (1)                                (2)
## -----
## Treatment_Dummy                0.612                            0.476
##                                (0.477)                            (0.325)
##
## TaskPhase1_Score                                1.374***
##                                (0.111)
##
## Constant                16.408***                            8.238***
##                                (0.423)                            (0.708)
## -----
## Observations                378                            378
## R2                0.004                            0.493
## Adjusted R2                0.002                            0.490
## Residual Std. Error  3.781 (df = 376)                2.702 (df = 375)
## F Statistic                1.591 (df = 1; 376) 182.159*** (df = 2; 375)
## =====
## Note:                                *p<0.1; **p<0.05; ***p<0.01
```

Task Phase 2 Analysis

```
# does any treatment have an effect on task phase 2 score?
mod_task2_a <- d_respondents[, lm(TaskPhase2_Score ~ Treatment_Dummy)]

mod_task2_b <- d_respondents[, lm(TaskPhase2_Score ~ Treatment_Dummy +
                                TaskPhase1_Score + as.factor(Gender) + as.factor(Education_Level) +
                                as.factor(Age_Range))]

stargazer(mod_task2_a,
           mod_task2_b,
           se = list(get_robust_se(mod_task2_a), get_robust_se(mod_task2_b)),
           omit = c("Education_Level", "Age_Range"),
           add.lines = list(c('Education Fixed Effects', 'No', 'Yes'),
                           c('Age Fixed Effects', 'No', 'Yes')),
           type='text')

##
## =====
##                                Dependent variable:
##                                -----
##                                TaskPhase2_Score
##                                (1)                                (2)
## -----
## Treatment_Dummy                0.577***                            0.511**
##                                (0.211)                            (0.217)
##
## TaskPhase1_Score                                0.331***
##                                (0.065)
##
## as.factor(Gender)Male                -0.041
##                                (0.172)
##
## Constant                4.553***                            2.051**
##                                (0.185)                            (0.852)
```

```
##
## -----
## Education Fixed Effects          No          Yes
## Age Fixed Effects               No          Yes
## Observations                   378         378
## R2                             0.018        0.172
## Adjusted R2                    0.015        0.142
## Residual Std. Error            1.736 (df = 376)    1.619 (df = 364)
## F Statistic                    6.699** (df = 1; 376) 5.815*** (df = 13; 364)
## =====
## Note:                          *p<0.1; **p<0.05; ***p<0.01
#does the specific treatment group have an effect on task phase 2 score?
mod_task2_c <- d_respondents[, lm(TaskPhase2_Score ~ as.factor(Assignment_Group))]

mod_task2_d <- d_respondents[, lm(TaskPhase2_Score ~ as.factor(Assignment_Group) +
                                TaskPhase1_Score + as.factor(Gender) + as.factor(Education_Level) +
                                Treatment_Dummy * as.factor(assign_bin))]

# Do you think that there are features of the data that might systematically predict that people will r
# TODO update this heterogeneity issue. I'm not quite sure this applies because they're both considered
# mod5 <- d_respondents[, lm(TaskPhase2_Score ~ Treatment_Dummy + as.factor(assign_bin) +
#                               Treatment_Dummy * as.factor(assign_bin))]
stargazer(mod_task2_c,
          mod_task2_d,
          se = list(get_robust_se(mod_task2_c), get_robust_se(mod_task2_d)),
          omit = c("Education_Level", "Age_Range"),
          add.lines = list(c('Education Fixed Effects', 'No', 'Yes'),
                           c('Age Fixed Effects', 'No', 'Yes')),
          type='text')

##
## =====
##                                     Dependent variable:
##                                     -----
##                                     TaskPhase2_Score
##                                     (1)                (2)
## -----
## as.factor(Assignment_Group)Medical Feedback    0.763***          0.602**
##                                                  (0.291)          (0.284)
##
## as.factor(Assignment_Group)Negative Images      0.356              0.400
##                                                  (0.254)          (0.254)
##
## as.factor(Assignment_Group)Positive Images      0.579**            0.514*
##                                                  (0.286)          (0.273)
##
## as.factor(Assignment_Group)Self-Reflect         0.612**            0.535*
##                                                  (0.272)          (0.288)
##
## TaskPhase1_Score                                0.327***
##                                                  (0.065)
##
## as.factor(Gender)Male                          -0.039
##                                                  (0.174)
##
```

```
## Constant                4.553***                2.085**
##                          (0.185)                (0.853)
## -----
## Education Fixed Effects                No                Yes
## Age Fixed Effects                No                Yes
## Observations                378                378
## R2                0.023                0.173
## Adjusted R2                0.013                0.137
## Residual Std. Error                1.738 (df = 373)                1.625 (df = 361)
## F Statistic                2.206* (df = 4; 373)                4.731*** (df = 16; 361)
## =====
## Note:                                *p<0.1; **p<0.05; ***p<0.01
```

Task Phase 3 Analysis

```
# test final task and any treatment
mod_task3_a <- d_respondents[, lm(TaskPhase3_Score ~ Treatment_Dummy)]
mod_task3_b <- d_respondents[, lm(TaskPhase3_Score ~ Treatment_Dummy +
                                TaskPhase1_Score + as.factor(Gender) + as.factor(Education_Level) +

stargazer(mod_task3_a,
           mod_task3_b,
           se = list(get_robust_se(mod_task3_a), get_robust_se(mod_task3_b)),
           omit = c("Education_Level", "Age_Range"),
           add.lines = list(c('Education Fixed Effects', 'No', 'Yes'),
                           c('Age Fixed Effects', 'No', 'Yes')),
           type='text')
```

```
##
## =====
##                               Dependent variable:
##                               -----
##                               TaskPhase3_Score
##                               (1)                (2)
## -----
## Treatment_Dummy                0.115                0.106
##                               (0.184)                (0.187)
##
## TaskPhase1_Score                0.213***
##                               (0.064)
##
## as.factor(Gender)Male                -0.011
##                               (0.165)
##
## Constant                5.316***                4.508***
##                               (0.156)                (0.753)
## -----
## Education Fixed Effects                No                Yes
## Age Fixed Effects                No                Yes
## Observations                378                378
## R2                0.001                0.100
## Adjusted R2                -0.002                0.068
```

```
## Residual Std. Error      1.638 (df = 376)      1.580 (df = 364)
## F Statistic              0.298 (df = 1; 376) 3.102*** (df = 13; 364)
## =====
## Note:                                *p<0.1; **p<0.05; ***p<0.01
```

```
# test final task and specific treatment
```

```
mod_task3_c <- d_respondents[, lm(TaskPhase3_Score ~ as.factor(Assignment_Group))]
```

```
mod_task3_d <- d_respondents[, lm(TaskPhase3_Score ~ as.factor(Assignment_Group) +
```

```
TaskPhase1_Score + as.factor(Gender) + as.factor(Education_Level) +
```

```
stargazer(mod_task3_c,
  mod_task3_d,
  se = list(get_robust_se(mod_task3_c),get_robust_se(mod_task3_d)),
  omit = c("Education_Level","Age_Range"),
  add.lines = list(c('Education Fixed Effects', 'No','Yes'),
    c('Age Fixed Effects','No','Yes')),
  type='text')
```

```
##
## =====
##                                     Dependent variable:
##                                     -----
##                                     TaskPhase3_Score
##                                     (1)                (2)
## -----
```

## as.factor(Assignment_Group)Medical Feedback	0.368	0.252
##	(0.264)	(0.255)
## as.factor(Assignment_Group)Negative Images	-0.043	0.035
##	(0.257)	(0.254)
## as.factor(Assignment_Group)Positive Images	0.026	0.058
##	(0.238)	(0.240)
## as.factor(Assignment_Group)Self-Reflect	0.109	0.083
##	(0.245)	(0.257)
## TaskPhase1_Score		0.208***
##		(0.062)
## as.factor(Gender)Male		-0.009
##		(0.166)
## Constant	5.316***	4.568***
##	(0.156)	(0.743)
## -----		
## Education Fixed Effects	No	Yes
## Age Fixed Effects	No	Yes
## Observations	378	378
## R2	0.008	0.102
## Adjusted R2	-0.003	0.062
## Residual Std. Error	1.638 (df = 373)	1.585 (df = 361)
## F Statistic	0.764 (df = 4; 373)	2.559*** (df = 16; 361)
## =====		

Note:

*p<0.1; **p<0.05; ***p<0.01

```
# qmplot(LocationLongitude, LocationLatitude, data = d_respondents, geom = "blank",
# zoom = 1, maptype = "toner-background", darken = .7, legend = "topleft")
# # ) +
# #   stat_density_2d(aes(fill = ..level..), geom = "polygon", alpha = .3, color = NA) +
# #   scale_fill_gradient2("Robbery\nPropensity", low = "white", mid = "yellow", high = "red", midpoint
# #
# d_respondents[, qmplot(LocationLongitude, LocationLatitude, geom="blank", zoom = 1)]
```