ETH Zurich June 2019

George Kinnear

18 June 2019

2PL results

This shows the results of fitting the 2PL model using the `mirt` package.

- a is discrimination
- b is difficulty
- endpoints of the 95% confidence intervals are also shown

Show 10 ▼ entries

Search:

a_CI_2.5	a_CI_97.5	a_value	b_CI_2.5	b_CI_97.5	b_value	Question
1.006	1.396	1.201	-3.357	-2.624	-2.990	A1
0.476	0.648	0.562	1.407	1.938	1.673	A2
1.230	1.467	1.348	-0.617	-0.468	-0.543	A3
1.068	1.286	1.177	-0.675	-0.509	-0.592	A4
1.340	1.584	1.462	0.015	0.143	0.079	A5
1.416	1.690	1.553	-0.943	-0.785	-0.864	A6
1.162	1.400	1.281	-0.986	-0.805	-0.896	A7
1.055	1.265	1.160	-0.007	0.139	0.066	A8
1.856	2.178	2.017	0.072	0.182	0.127	A9
1.620	1.904	1.762	-0.206	-0.089	-0.147	A10
a_CI_2.5	a_CI_97.5	a_value	b_CI_2.5	b_CI_97.5	b_value	Question

Showing 1 to 10 of 36 entries

Previous 1 2 3 4 Next

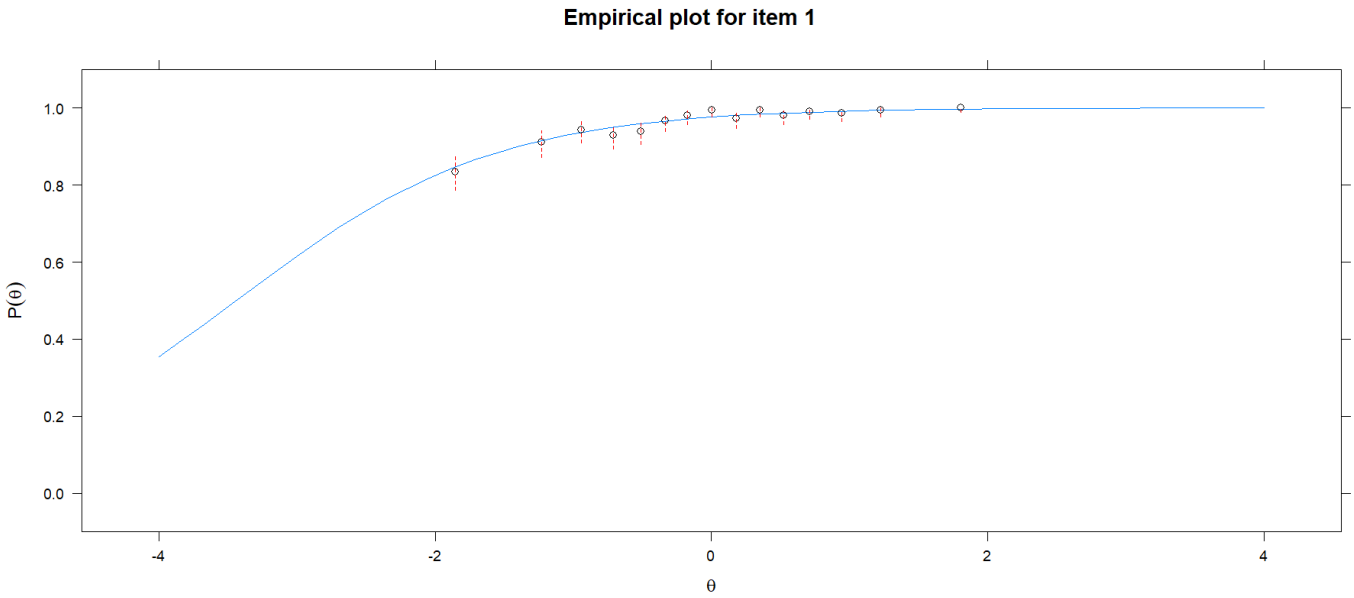
Item Fit

Comparison of the IRT curve and the observed proportions.

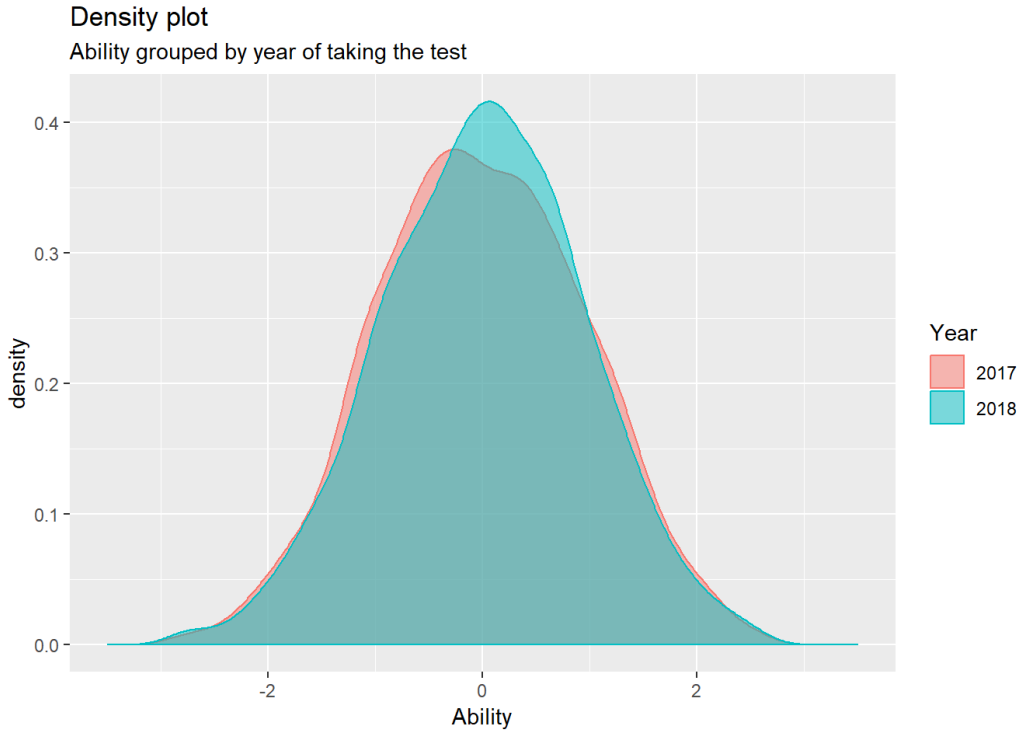
Item:

1 36

1 5 9 13 17 21 25 29 33 36

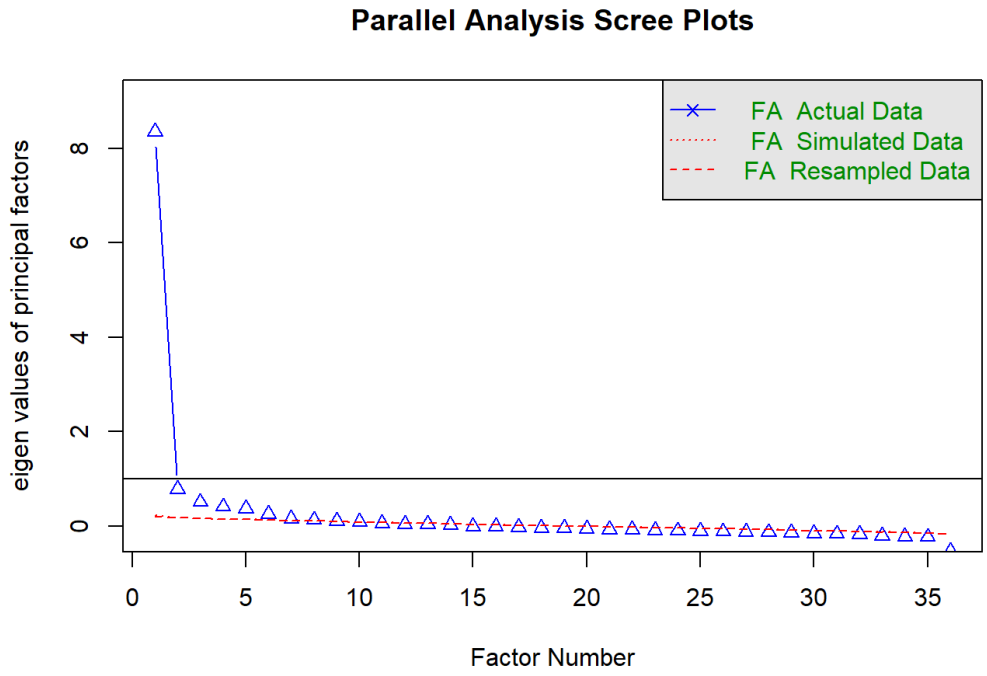


Comparing 2017 and 2018



Factor analysis

```
library(psych)
fa.parallel(eth_both_years[2:37], fa='fa')
```



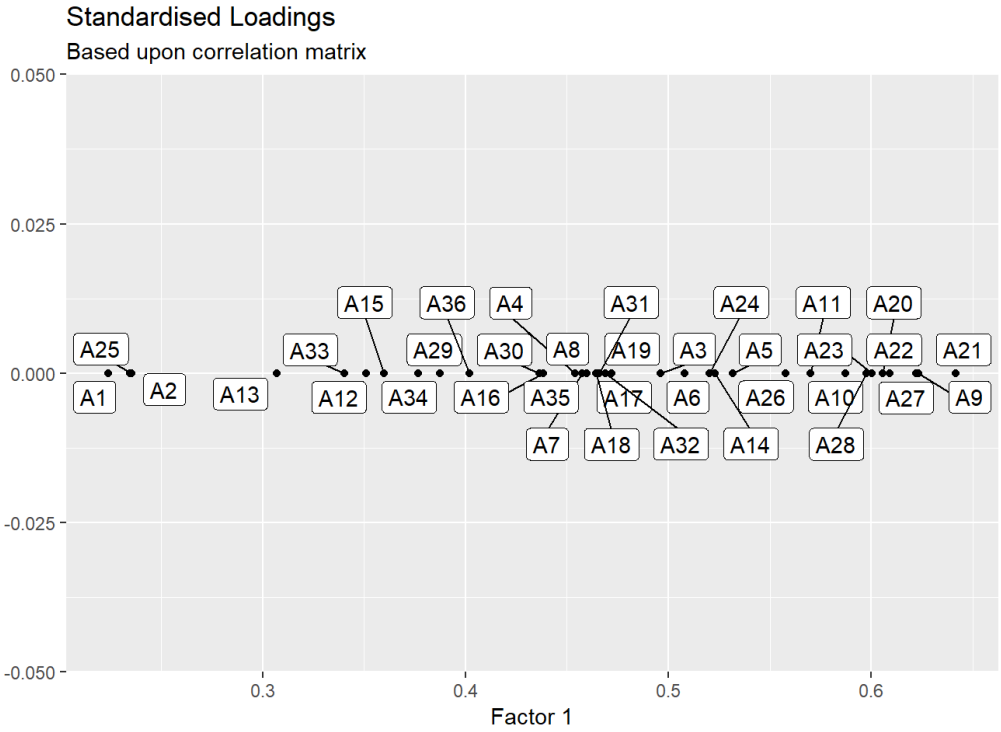

```
## Parallel analysis suggests that the number of factors =  8  and the number of components =  NA
```

1 Factor

```
fitfact <- factanal(eth_both_years[2:37], 1, rotation="varimax")
print(fitfact, digits=2, cutoff=.3, sort=TRUE)
```

```
##
## Call:
## factanal(x = eth_both_years[2:37], factors = 1, rotation = "varimax")
##
## Uniquenesses:
##   A1   A2   A3   A4   A5   A6   A7   A8   A9  A10  A11  A12  A13  A14  A15
## 0.95 0.95 0.75 0.79 0.72 0.74 0.79 0.79 0.61 0.66 0.67 0.88 0.91 0.73 0.87
##  A16  A17  A18  A19  A20  A21  A22  A23  A24  A25  A26  A27  A28  A29  A30
## 0.81 0.78 0.78 0.78 0.63 0.59 0.61 0.64 0.73 0.94 0.69 0.63 0.64 0.85 0.81
##  A31  A32  A33  A34  A35  A36
## 0.78 0.78 0.88 0.86 0.79 0.84
##
## Loadings:
##  [1] 0.53 0.51 0.62 0.59 0.57 0.52 0.61 0.64 0.62 0.60 0.52 0.56 0.61 0.60
## [15]           0.50 0.45 0.46 0.46 0.35 0.31 0.36 0.44 0.47 0.46 0.47
## [29] 0.39 0.44 0.47 0.47 0.34 0.38 0.46 0.40
##
##                Factor1
## SS loadings       8.35
## Proportion Var    0.23
##
## Test of the hypothesis that 1 factor is sufficient.
## The chi square statistic is 5428.36 on 594 degrees of freedom.
## The p-value is 0
```

```
load <- data.frame(fitfact$loadings[,])
names(load) = c("Factor1")
ggplot(load, aes(x=Factor1, y=0)) +
  geom_point() +
  geom_label_repel(aes(label = rownames(load)), show.legend=F) +
  labs(x='Factor 1',
       y='',
       title='Standardised Loadings',
       subtitle='Based upon correlation matrix')
```

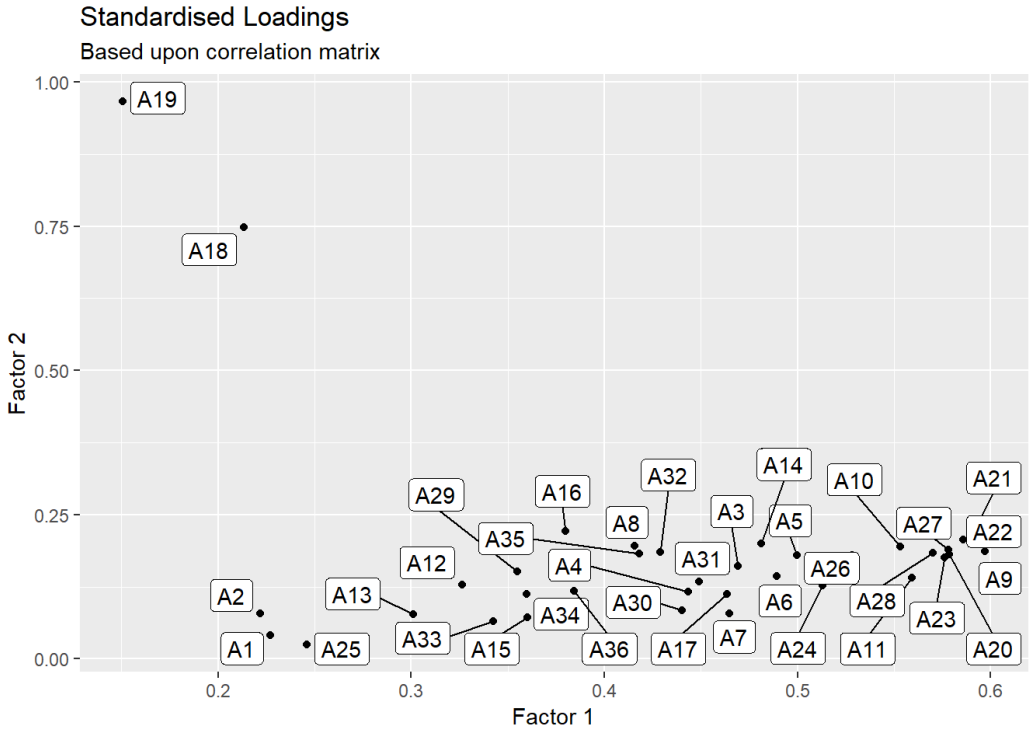


2 Factors

```
fitfact <- factanal(eth_both_years[2:37], 2, rotation="varimax")
print(fitfact, digits=2, cutoff=.3, sort=TRUE)
```

```
##
## Call:
## factanal(x = eth_both_years[2:37], factors = 2, rotation = "varimax")
##
## Uniquenesses:
##   A1   A2   A3   A4   A5   A6   A7   A8   A9  A10  A11  A12  A13  A14  A15
## 0.95 0.94 0.75 0.79 0.72 0.74 0.78 0.79 0.61 0.66 0.67 0.88 0.90 0.73 0.87
##  A16  A17  A18  A19  A20  A21  A22  A23  A24  A25  A26  A27  A28  A29  A30
## 0.81 0.77 0.40 0.04 0.63 0.59 0.61 0.64 0.72 0.94 0.69 0.63 0.64 0.85 0.80
##  A31  A32  A33  A34  A35  A36
## 0.78 0.78 0.88 0.86 0.79 0.84
##
## Loadings:
##     Factor1 Factor2
## A9  0.60
## A10 0.55
## A11 0.56
## A20 0.58
## A21 0.59
## A22 0.59
## A23 0.58
## A24 0.51
## A26 0.53
## A27 0.58
## A28 0.57
## A18         0.75
## A19         0.97
## A1
## A2
## A3  0.47
## A4  0.44
## A5  0.50
## A6  0.49
## A7  0.46
## A8  0.42
## A12 0.33
## A13 0.30
## A14 0.48
## A15 0.36
## A16 0.38
## A17 0.46
## A25
## A29 0.36
## A30 0.44
## A31 0.45
## A32 0.43
## A33 0.34
## A34 0.36
## A35 0.42
## A36 0.38
##
##                Factor1 Factor2
## SS loadings       7.25    2.29
## Proportion Var    0.20    0.06
## Cumulative Var    0.20    0.27
##
## Test of the hypothesis that 2 factors are sufficient.
## The chi square statistic is 3050.55 on 559 degrees of freedom.
## The p-value is 0
```

```
load <- data.frame(fitfact$loadings[,])
ggplot(load, aes(x=Factor1, y=Factor2)) +
  geom_point() +
  geom_label_repel(aes(label = rownames(load)), show.legend=F) +
  labs(x='Factor 1',
       y='Factor 2',
       title='Standardised Loadings',
       subtitle='Based upon correlation matrix')
```



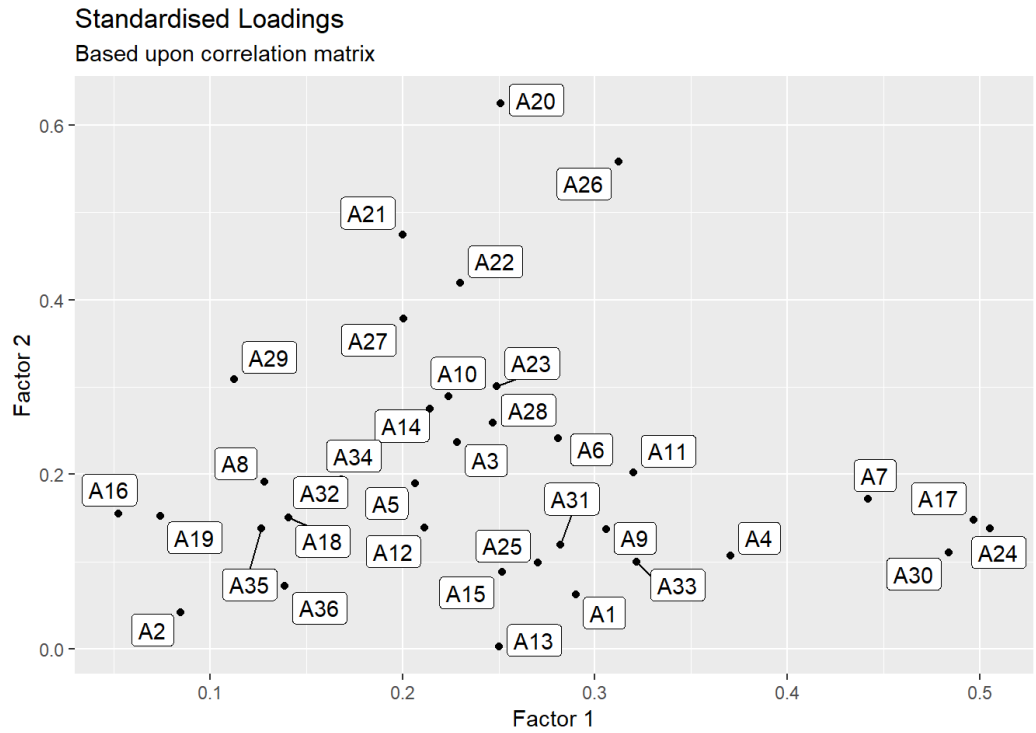
9 Factors (showing only first 2)

```
fitfact <- factanal(eth_both_years[2:37], 9, rotation="varimax")
print(fitfact, digits=2, cutoff=.3, sort=TRUE)
```

```
##
## Call:
## factanal(x = eth_both_years[2:37], factors = 9, rotation = "varimax")
##
## Uniquenesses:
##   A1   A2   A3   A4   A5   A6   A7   A8   A9  A10  A11  A12  A13  A14  A15
## 0.88 0.64 0.71 0.74 0.64 0.66 0.72 0.74 0.46 0.64 0.64 0.86 0.79 0.69 0.82
##  A16  A17  A18  A19  A20  A21  A22  A23  A24  A25  A26  A27  A28  A29  A30
## 0.56 0.66 0.41 0.00 0.47 0.51 0.57 0.60 0.62 0.91 0.51 0.59 0.61 0.80 0.71
##  A31  A32  A33  A34  A35  A36
## 0.75 0.74 0.83 0.69 0.55 0.70
##
## Loadings:
##     Factor1 Factor2 Factor3 Factor4 Factor5 Factor6 Factor7 Factor8
## A24  0.51
## A20          0.62
## A26  0.31    0.56
## A9   0.31            0.59
## A18                          0.70
## A19                          0.96
## A35                                  0.61
## A16                                          0.58
## A2                                                   0.59
## A1
## A3                                                           0.33
## A4   0.37
## A5                                                           0.38
## A6                                                           0.36
## A7   0.44
## A8                   0.31
## A10                  0.35
## A11  0.32            0.39
## A12
## A13                                                  0.34
## A14
## A15
## A17  0.50
## A21          0.47
## A22          0.42
## A23          0.30    0.41
## A25
## A27          0.38
## A28                  0.43
## A29          0.31
## A30  0.48
## A31
## A32
## A33  0.32
## A34                                  0.47
## A36                                  0.44
##     Factor9
## A24
## A20
## A26
## A9
## A18
## A19
## A35
## A16
## A2
## A1
## A3
## A4
## A5
## A6
## A7
## A8
## A10
## A11
## A12
## A13
## A14
## A15
## A17
```

```
## A21
## A22
## A23
## A25
## A27
## A28
## A29
## A30
## A31
## A32
## A33
## A34
## A36
##
##                Factor1 Factor2 Factor3 Factor4 Factor5 Factor6 Factor7
## SS loadings       2.61    2.22    1.93    1.69    1.37    0.96    0.81
## Proportion Var    0.07    0.06    0.05    0.05    0.04    0.03    0.02
## Cumulative Var    0.07    0.13    0.19    0.23    0.27    0.30    0.32
##                Factor8 Factor9
## SS loadings       0.76    0.24
## Proportion Var    0.02    0.01
## Cumulative Var    0.34    0.35
##
## Test of the hypothesis that 9 factors are sufficient.
## The chi square statistic is 528.49 on 342 degrees of freedom.
## The p-value is 3.63e-10
```

```
load <- data.frame(fitfact$loadings[,])
ggplot(load, aes(x=Factor1, y=Factor2)) +
  geom_point() +
  geom_label_repel(aes(label = rownames(load)), show.legend=F) +
  labs(x='Factor 1',
       y='Factor 2',
       title='Standardised Loadings',
       subtitle='Based upon correlation matrix')
```

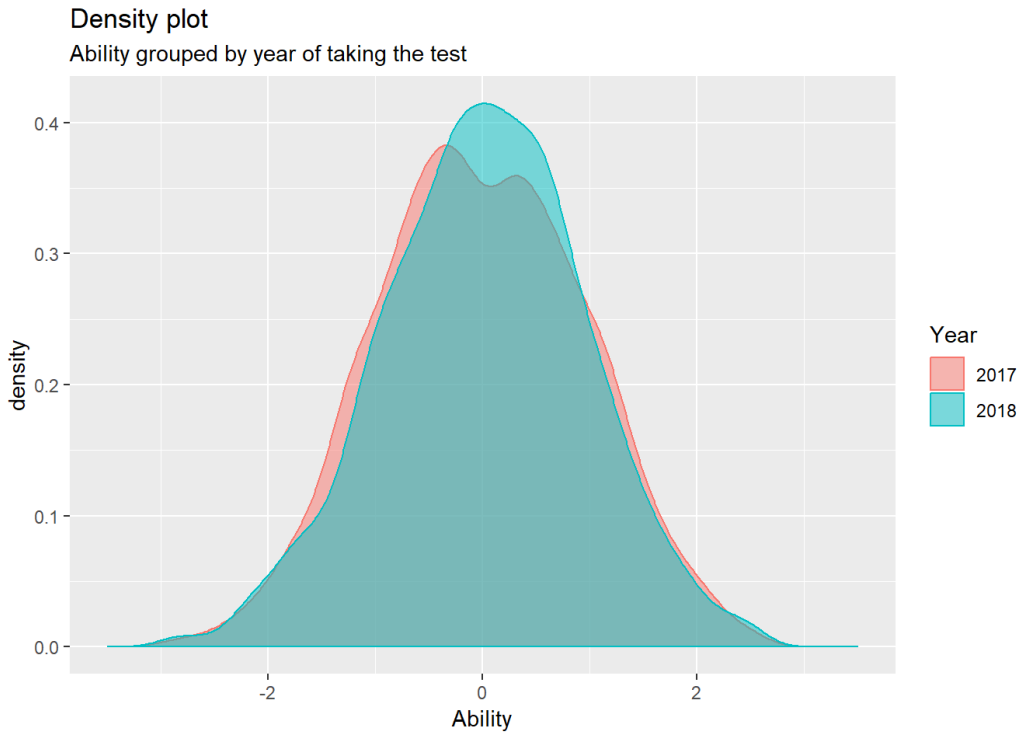


Investigating differences between groups

Do students from different programmes of study have different distributions of ability?

Differences between year groups

```
# Compare the distribution of abilities in the two years
ggplot(eth_entry_test, aes(F1, fill=as.factor(year), colour=as.factor(year))) +
  geom_density(alpha=0.5) +
  scale_x_continuous(limits=c(-3.5,3.5)) +
  labs(title="Density plot",
       subtitle="Ability grouped by year of taking the test",
       x="Ability",
       fill='Year', colour='Year')
```



There does not seem to be a big difference between the two year groups, so we combine them in the following analysis.

Differences between classes

```
# Compare the distribution of abilities in the two years
ggplot(eth_entry_test, aes(F1, fill=as.factor(class), colour=as.factor(class))) +
  geom_density(alpha=0.5) +
  scale_x_continuous(limits=c(-3.5,3.5)) +
  labs(title="Density plot",
       subtitle="Ability grouped by class",
       x="Ability",
       fill='Class', colour='Class')
```

That plot is hard to read, so try another approach:

```
library(ggridges)

# Compare the distribution of abilities in the two years
ggplot(eth_entry_test, aes(x = F1, y = class, colour = class, fill = class)) +
  geom_density_ridges(alpha=0.5) +
  scale_x_continuous(limits=c(-3.5,3.5)) +
  labs(title="Density plot",
       subtitle="Ability grouped by class",
       x="Ability",
       fill='Class', colour='Class')
```

```
## Picking joint bandwidth of 0.256
```