STATS 3001 / STATS 4104 / STATS 7054
Statistical Modelling III
Practical 6 - Model selection - solutions

Week 11

Data

Excel spreadsheet has NA as character to let R know

```
marks <- readxl::read_excel(here::here("data","marks.xlsx"), na = "NA")
marks
```

```
## # A tibble: 350 x 8
##       OQ    A1    A2    A3    A4    A5    A6     E
##    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
##  1 4.95     42    36    32    40    27    30    64
##  2 0.769    36    30    31    31    21    21     0
##  3 4.62     41    35    31    39    28    32    52
##  4 2.38     32    19    31    35    26    27    51
##  5 4.03     44    39    18    38    26    31    60
##  6 2.75     38    33    30    34    23    29    45
##  7 5        43    35    35    31    27    34    51
##  8 5        44    36    35    40    27    34    63
##  9 4.91     44    34    35    39    28    30    54
## 10 5        44    34    35    41    28    34    60
## # ... with 340 more rows
```

EDA

```
skimr::skim_without_charts(marks)
```

Table 1: Data summary

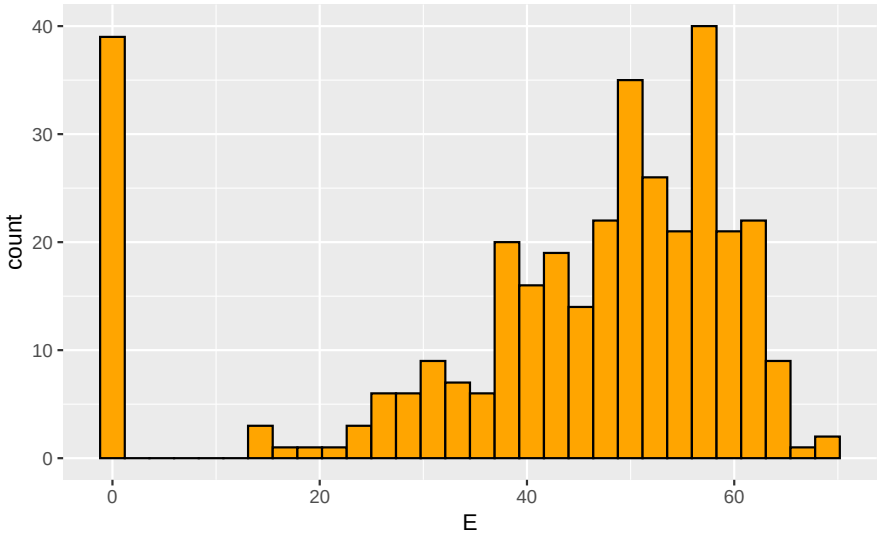
Name	marks
Number of rows	350
Number of columns	8
Column type frequency:	
numeric	8
Group variables	None

Variable type: numeric

skim_variable	n_missing	complete_rate	mean	sd	p0	p25	p50	p75	p100
OQ	0	1.00	3.70	1.70	0	3	4.56	4.97	5
A1	11	0.97	37.35	8.22	0	35	39.00	42.50	45
A2	11	0.97	30.31	8.19	0	28	32.00	35.50	41
A3	11	0.97	26.01	8.72	0	23	29.00	32.00	35
A4	11	0.97	31.16	11.57	0	30	35.00	38.00	41
A5	11	0.97	20.30	9.05	0	20	24.00	26.00	29
A6	11	0.97	23.41	10.36	0	21	27.00	30.25	34
E	0	1.00	43.04	18.40	0	38	49.00	56.00	69

```
marks %>%
  ggplot(aes(E)) +
  geom_histogram(col = "black", fill = "orange")
```

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

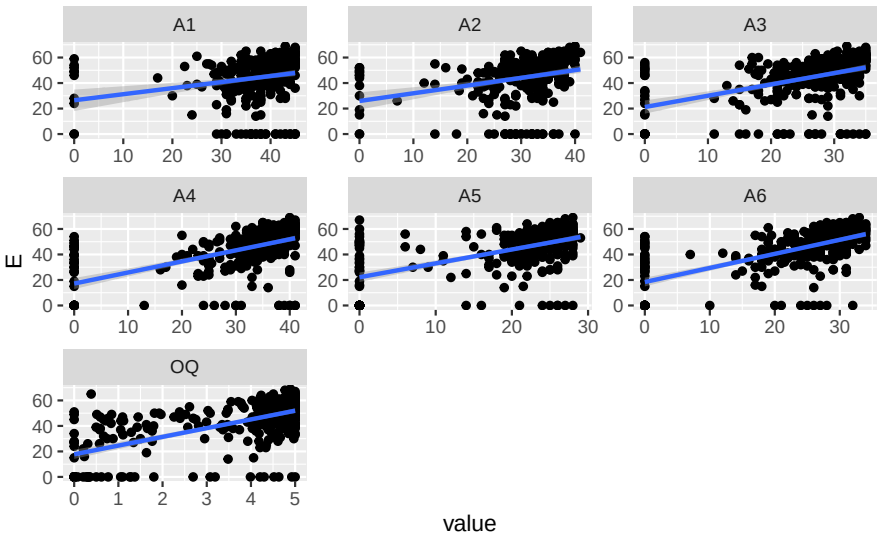



```
marks %>%
  pivot_longer(-E) %>%
  ggplot(aes(value, E)) +
  geom_point() +
  facet_wrap(~name, scales = "free") +
  geom_smooth(method = lm)
```

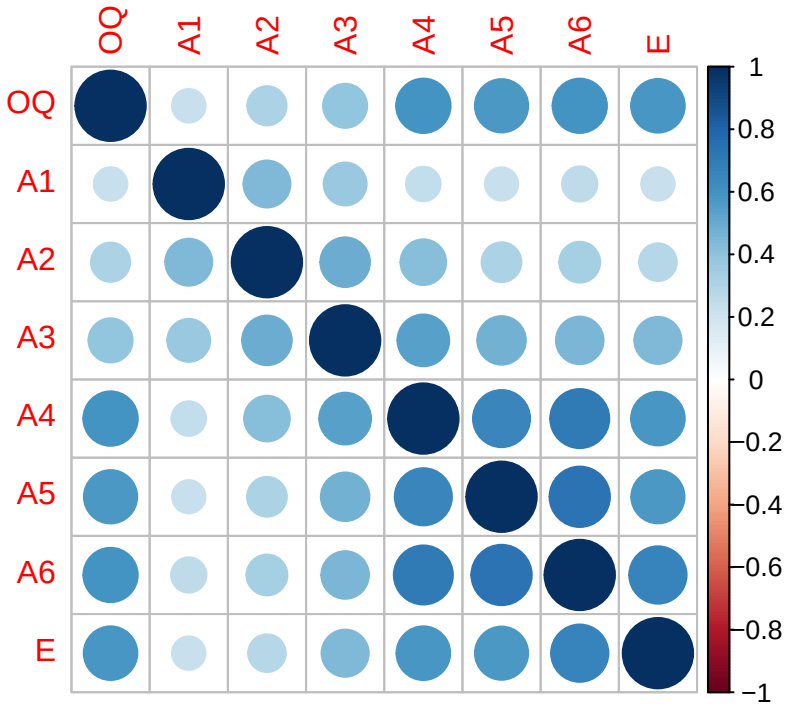
```
## `geom_smooth()` using formula 'y ~ x'
```

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

```
## Warning: Removed 66 rows containing missing values (geom_point).
```




```
marks %>%
  na.omit() %>%
  select(where(is.numeric)) %>%
  cor() %>%
  corrplot::corrplot()
```



Cleaning

Going to set NA as zero as no submitted work

```
set_NA_zero <- function(x){
  x[is.na(x)] <- 0
  return(x)
}
marks <-
  marks %>%
  mutate(
  across(where(is.numeric), set_NA_zero)
)
marks
```

```
## # A tibble: 350 x 8
##       OQ    A1    A2    A3    A4    A5    A6     E
##    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
##  1 4.95     42    36    32    40    27    30    64
##  2 0.769    36    30    31    31    21    21     0
##  3 4.62     41    35    31    39    28    32    52
##  4 2.38     32    19    31    35    26    27    51
##  5 4.03     44    39    18    38    26    31    60
##  6 2.75     38    33    30    34    23    29    45
##  7 5        43    35    35    31    27    34    51
##  8 5        44    36    35    40    27    34    63
##  9 4.91     44    34    35    39    28    30    54
## 10 5        44    34    35    41    28    34    60
## # ... with 340 more rows
```

Decided to have on similar scale, so scale by max value as do not have total for each part.

```
scale <- function(x){
  x <- x / max(x, na.rm = TRUE)
  return(x)
}
marks <-
  marks %>%
  mutate(
  across(where(is.numeric), scale)
)
marks
```

```
## # A tibble: 350 x 8
##       OQ    A1    A2    A3    A4    A5    A6     E
##    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl>
##  1 0.989 0.933 0.878 0.914 0.976 0.931 0.882 0.928
##  2 0.154 0.8   0.732 0.886 0.756 0.724 0.618 0
##  3 0.923 0.911 0.854 0.886 0.951 0.966 0.941 0.754
##  4 0.477 0.711 0.463 0.886 0.854 0.897 0.794 0.739
##  5 0.805 0.978 0.951 0.514 0.927 0.897 0.912 0.870
##  6 0.550 0.844 0.805 0.857 0.829 0.793 0.853 0.652
##  7 1     0.956 0.854 1     0.756 0.931 1     0.739
##  8 1     0.978 0.878 1     0.976 0.931 1     0.913
##  9 0.983 0.978 0.829 1     0.951 0.966 0.882 0.783
## 10 1     0.978 0.829 1     1     0.966 1     0.870
## # ... with 340 more rows
```

Backwards using P-value

```
full <- lm(E ~ ., data = marks)
summary(full)
```

```
##
## Call:
## lm(formula = E ~ ., data = marks)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.82520 -0.07504  0.02280  0.09070  0.63614
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.08970    0.03617   2.480   0.0136 *
## OQ           0.19728    0.03804   5.186 3.69e-07 ***
## A1           0.04474    0.05452   0.821   0.4124
## A2          -0.03313    0.05846  -0.567   0.5713
## A3           0.12601    0.05006   2.517   0.0123 *
## A4           0.07666    0.05396   1.421   0.1563
## A5           0.03950    0.04818   0.820   0.4129
## A6           0.30281    0.05206   5.817 1.38e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1749 on 342 degrees of freedom
## Multiple R-squared:  0.5787, Adjusted R-squared:   0.57
## F-statistic:  67.1 on 7 and 342 DF,  p-value: < 2.2e-16
```

```
M2 <- update(full, . ~ . - A2)
summary(M2)
```

```
##
## Call:
## lm(formula = E ~ OQ + A1 + A3 + A4 + A5 + A6, data = marks)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.82765 -0.07461  0.02407  0.08899  0.62425
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.08610    0.03557   2.420   0.0160 *
## OQ           0.19678    0.03800   5.179 3.81e-07 ***
## A1           0.03159    0.04928   0.641   0.5219
## A3           0.11813    0.04804   2.459   0.0144 *
## A4           0.07132    0.05308   1.344   0.1799
## A5           0.03991    0.04813   0.829   0.4076
## A6           0.30302    0.05200   5.827 1.30e-08 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Residual standard error: 0.1747 on 343 degrees of freedom
## Multiple R-squared:  0.5783, Adjusted R-squared:  0.5709
## F-statistic: 78.39 on 6 and 343 DF,  p-value: < 2.2e-16
```

```
M3 <- update(M2, . ~ . - A1)
summary(M3)
```

```
##
## Call:
## lm(formula = E ~ OQ + A3 + A4 + A5 + A6, data = marks)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.82593 -0.07362  0.02514  0.09000  0.63920
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.09993    0.02827   3.535 0.000463 ***
## OQ           0.19798    0.03792   5.221 3.08e-07 ***
## A3           0.12957    0.04457   2.907 0.003883 **
## A4           0.07261    0.05300   1.370 0.171537
## A5           0.03981    0.04809   0.828 0.408303
## A6           0.30536    0.05183   5.892 9.12e-09 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1746 on 344 degrees of freedom
## Multiple R-squared:  0.5778, Adjusted R-squared:  0.5716
## F-statistic: 94.14 on 5 and 344 DF,  p-value: < 2.2e-16
```

```
M4 <- update(M3, . ~ . - A5)
summary(M4)
```

```
##
## Call:
## lm(formula = E ~ OQ + A3 + A4 + A6, data = marks)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.82490 -0.07517  0.02499  0.09198  0.63908
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.10005    0.02825   3.541 0.000453 ***
## OQ           0.20308    0.03740   5.431 1.06e-07 ***
## A3           0.13507    0.04405   3.066 0.002340 **
## A4           0.08099    0.05200   1.557 0.120280
## A6           0.32482    0.04617   7.035 1.07e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1745 on 345 degrees of freedom
```

```
## Multiple R-squared:  0.5769, Adjusted R-squared:  0.572
## F-statistic: 117.6 on 4 and 345 DF,  p-value: < 2.2e-16
```

```
PV_back <- update(M4, . ~ . - A4)
summary(PV_back)
```

```
##
## Call:
## lm(formula = E ~ OQ + A3 + A6, data = marks)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.82442 -0.06864  0.02412  0.09377  0.63288
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.10625    0.02803   3.791 0.000177 ***
## OQ           0.21881    0.03608   6.064 3.47e-09 ***
## A3           0.16017    0.04108   3.899 0.000116 ***
## A6           0.36040    0.04021   8.964  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1748 on 346 degrees of freedom
## Multiple R-squared:  0.574,  Adjusted R-squared:  0.5703
## F-statistic: 155.4 on 3 and 346 DF,  p-value: < 2.2e-16
```

Backwards using AIC

```
AIC_back <- stats::step(full)
```

```
## Start:  AIC=-1212.64
## E ~ OQ + A1 + A2 + A3 + A4 + A5 + A6
##
##        Df Sum of Sq    RSS     AIC
## - A2    1   0.00982 10.470 -1214.3
## - A5    1   0.02056 10.480 -1214.0
## - A1    1   0.02060 10.480 -1214.0
## <none>              10.460 -1212.6
## - A4    1   0.06172 10.521 -1212.6
## - A3    1   0.19380 10.654 -1208.2
## - OQ    1   0.82243 11.282 -1188.2
## - A6    1   1.03487 11.495 -1181.6
##
## Step:  AIC=-1214.31
## E ~ OQ + A1 + A3 + A4 + A5 + A6
##
##        Df Sum of Sq    RSS     AIC
## - A1    1   0.01254 10.482 -1215.9
## - A5    1   0.02099 10.491 -1215.6
## - A4    1   0.05511 10.525 -1214.5
```

```
## <none>              10.470 -1214.3
## - A3    1   0.18456 10.654 -1210.2
## - OQ    1   0.81872 11.288 -1190.0
## - A6    1   1.03635 11.506 -1183.3
##
## Step:  AIC=-1215.89
## E ~ OQ + A3 + A4 + A5 + A6
##
##        Df Sum of Sq    RSS     AIC
## - A5    1   0.02089 10.503 -1217.2
## - A4    1   0.05720 10.539 -1216.0
## <none>              10.482 -1215.9
## - A3    1   0.25754 10.740 -1209.4
## - OQ    1   0.83073 11.313 -1191.2
## - A6    1   1.05767 11.540 -1184.2
##
## Step:  AIC=-1217.19
## E ~ OQ + A3 + A4 + A6
##
##        Df Sum of Sq    RSS     AIC
## <none>              10.503 -1217.2
## - A4    1   0.07385 10.577 -1216.7
## - A3    1   0.28619 10.789 -1209.8
## - OQ    1   0.89784 11.401 -1190.5
## - A6    1   1.50686 12.010 -1172.3
```

```
summary(AIC_back)
```

```
##
## Call:
## lm(formula = E ~ OQ + A3 + A4 + A6, data = marks)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.82490 -0.07517  0.02499  0.09198  0.63908
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.10005    0.02825   3.541 0.000453 ***
## OQ           0.20308    0.03740   5.431 1.06e-07 ***
## A3           0.13507    0.04405   3.066 0.002340 **
## A4           0.08099    0.05200   1.557 0.120280
## A6           0.32482    0.04617   7.035 1.07e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1745 on 345 degrees of freedom
## Multiple R-squared:  0.5769, Adjusted R-squared:  0.572
## F-statistic: 117.6 on 4 and 345 DF,  p-value: < 2.2e-16
```

Forwards using AIC

```
null <- lm(E ~ 1, data = marks)
AIC_forward <- stats::step(
  null,
  scope = E ~ OQ + A1 + A2 + A3 + A4 + A5 + A6,
  direction = "forward"
)
```

```
## Start:  AIC=-924.12
## E ~ 1
##
##        Df Sum of Sq    RSS      AIC
## + A6    1   12.3244 12.501 -1162.24
## + A4    1   10.2554 14.570 -1108.64
## + OQ    1    9.9424 14.883 -1101.20
## + A5    1    9.7112 15.114 -1095.80
## + A3    1    7.1616 17.664 -1041.24
## + A2    1    4.4343 20.391  -990.99
## + A1    1    3.8327 20.993  -980.81
## <none>              24.826  -924.12
##
## Step:  AIC=-1162.24
## E ~ A6
##
##        Df Sum of Sq    RSS     AIC
## + OQ    1   1.45964 11.041 -1203.7
## + A3    1   0.80005 11.701 -1183.4
## + A4    1   0.73954 11.762 -1181.6
## + A5    1   0.40887 12.092 -1171.9
## + A1    1   0.30065 12.200 -1168.8
## + A2    1   0.28193 12.219 -1168.2
## <none>              12.501 -1162.2
##
## Step:  AIC=-1203.69
## E ~ A6 + OQ
##
##        Df Sum of Sq    RSS     AIC
## + A3    1   0.46466 10.577 -1216.7
## + A4    1   0.25232 10.789 -1209.8
## + A1    1   0.15225 10.889 -1206.5
## + A5    1   0.11823 10.923 -1205.5
## + A2    1   0.11010 10.931 -1205.2
## <none>              11.041 -1203.7
##
## Step:  AIC=-1216.74
## E ~ A6 + OQ + A3
##
##        Df Sum of Sq    RSS     AIC
## + A4    1  0.073845 10.503 -1217.2
## <none>              10.577 -1216.7
## + A5    1  0.037528 10.539 -1216.0
## + A1    1  0.014832 10.562 -1215.2
```

```
## + A2    1  0.000015 10.577 -1214.7
##
## Step:  AIC=-1217.19
## E ~ A6 + OQ + A3 + A4
##
##        Df Sum of Sq    RSS     AIC
## <none>              10.503 -1217.2
## + A5    1 0.0208854 10.482 -1215.9
## + A1    1 0.0124414 10.491 -1215.6
## + A2    1 0.0019454 10.501 -1215.3
```

```
summary(AIC_forward)
```

```
##
## Call:
## lm(formula = E ~ A6 + OQ + A3 + A4, data = marks)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.82490 -0.07517  0.02499  0.09198  0.63908
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.10005    0.02825   3.541 0.000453 ***
## A6           0.32482    0.04617   7.035 1.07e-11 ***
## OQ           0.20308    0.03740   5.431 1.06e-07 ***
## A3           0.13507    0.04405   3.066 0.002340 **
## A4           0.08099    0.05200   1.557 0.120280
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1745 on 345 degrees of freedom
## Multiple R-squared:  0.5769, Adjusted R-squared:  0.572
## F-statistic: 117.6 on 4 and 345 DF,  p-value: < 2.2e-16
```

Both using AIC

```
null <- lm(E ~ 1, data = marks)
AIC_both <- stats::step(
  null,
  scope = E ~ OQ + A1 + A2 + A3 + A4 + A5 + A6,
  direction = "both"
)
```

```
## Start:  AIC=-924.12
## E ~ 1
##
##        Df Sum of Sq    RSS      AIC
## + A6    1   12.3244 12.501 -1162.24
## + A4    1   10.2554 14.570 -1108.64
## + OQ    1    9.9424 14.883 -1101.20
```

```
## + A5    1    9.7112 15.114 -1095.80
## + A3    1    7.1616 17.664 -1041.24
## + A2    1    4.4343 20.391  -990.99
## + A1    1    3.8327 20.993  -980.81
## <none>              24.826  -924.12
##
## Step:  AIC=-1162.24
## E ~ A6
##
##        Df Sum of Sq    RSS      AIC
## + OQ    1    1.4596 11.041 -1203.69
## + A3    1    0.8001 11.701 -1183.39
## + A4    1    0.7395 11.762 -1181.58
## + A5    1    0.4089 12.092 -1171.88
## + A1    1    0.3006 12.200 -1168.76
## + A2    1    0.2819 12.219 -1168.22
## <none>              12.501 -1162.24
## - A6    1   12.3244 24.826  -924.12
##
## Step:  AIC=-1203.69
## E ~ A6 + OQ
##
##        Df Sum of Sq    RSS     AIC
## + A3    1    0.4647 10.577 -1216.7
## + A4    1    0.2523 10.789 -1209.8
## + A1    1    0.1523 10.889 -1206.5
## + A5    1    0.1182 10.923 -1205.5
## + A2    1    0.1101 10.931 -1205.2
## <none>              11.041 -1203.7
## - OQ    1    1.4596 12.501 -1162.2
## - A6    1    3.8416 14.883 -1101.2
##
## Step:  AIC=-1216.74
## E ~ A6 + OQ + A3
##
##        Df Sum of Sq    RSS     AIC
## + A4    1   0.07385 10.503 -1217.2
## <none>              10.577 -1216.7
## + A5    1   0.03753 10.539 -1216.0
## + A1    1   0.01483 10.562 -1215.2
## + A2    1   0.00002 10.577 -1214.7
## - A3    1   0.46466 11.041 -1203.7
## - OQ    1   1.12425 11.701 -1183.4
## - A6    1   2.45617 13.033 -1145.7
##
## Step:  AIC=-1217.19
## E ~ A6 + OQ + A3 + A4
##
##        Df Sum of Sq    RSS     AIC
## <none>              10.503 -1217.2
## - A4    1   0.07385 10.577 -1216.7
## + A5    1   0.02089 10.482 -1215.9
## + A1    1   0.01244 10.491 -1215.6
## + A2    1   0.00195 10.501 -1215.3
```

```
## - A3    1   0.28619 10.789 -1209.8
## - OQ    1   0.89784 11.401 -1190.5
## - A6    1   1.50686 12.010 -1172.3
```

```
summary(AIC_both)
```

```
##
## Call:
## lm(formula = E ~ A6 + OQ + A3 + A4, data = marks)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.82490 -0.07517  0.02499  0.09198  0.63908
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.10005    0.02825   3.541 0.000453 ***
## A6           0.32482    0.04617   7.035 1.07e-11 ***
## OQ           0.20308    0.03740   5.431 1.06e-07 ***
## A3           0.13507    0.04405   3.066 0.002340 **
## A4           0.08099    0.05200   1.557 0.120280
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1745 on 345 degrees of freedom
## Multiple R-squared:  0.5769, Adjusted R-squared:  0.572
## F-statistic: 117.6 on 4 and 345 DF,  p-value: < 2.2e-16
```

```
stargazer(PV_back, AIC_back, AIC_forward, AIC_both, type = "text")
```


Dependent variable:				
	E			
	(1)	(2)	(3)	(4)
OQ	0.219***	0.203***	0.203***	0.203***
	(0.036)	(0.037)	(0.037)	(0.037)
A3	0.160***	0.135***	0.135***	0.135***
	(0.041)	(0.044)	(0.044)	(0.044)
A4	0.081	0.081	0.081	
	(0.052)	(0.052)	(0.052)	
A6	0.360***	0.325***	0.325***	0.325***
	(0.040)	(0.046)	(0.046)	(0.046)
Constant	0.106***	0.100***	0.100***	0.100***
	(0.028)	(0.028)	(0.028)	(0.028)
Observations	350	350	350	350
R2	0.574	0.577	0.577	0.577

Adjusted R2 0.570 0.572 0.572 0.572
Residual Std. Error 0.175 (df = 346) 0.174 (df = 345) 0.174 (df = 345) 0.174 (df = 345)
F Statistic 155.372*** (df = 3; 346) 117.616*** (df = 4; 345) 117.616*** (df = 4; 345) 117.616*** (df = 4; 345)
==========
Note: *p<0.1;* ***p<0.05;*** p<0.01

So we have two models:

```
Model 1: E ~ OQ + A3 + A4 + A6
```

```
Model 2: E ~ OQ + A3 + A6
```

We will decide between them using 5-fold CV with RSME.

CV

Split data

```
folds <- rep(1:5, each = 70)
folds
```

```
##   [1] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
##  [38] 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 2 2 2 2 2
##  [75] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
## [112] 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 3 3 3 3 3 3 3 3
## [149] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3
## [186] 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 4 4 4 4 4 4 4 4 4 4 4
## [223] 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4
## [260] 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 4 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [297] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
## [334] 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5
```

```
marks_CV <-
  marks %>%
  add_column(
    fold = sample(folds)
    )
marks_CV
```

```
## # A tibble: 350 x 9
##       OQ    A1    A2    A3    A4    A5    A6     E  fold
##    <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <dbl> <int>
##  1 0.989 0.933 0.878 0.914 0.976 0.931 0.882 0.928     5
##  2 0.154 0.8   0.732 0.886 0.756 0.724 0.618 0         3
##  3 0.923 0.911 0.854 0.886 0.951 0.966 0.941 0.754     3
##  4 0.477 0.711 0.463 0.886 0.854 0.897 0.794 0.739     5
##  5 0.805 0.978 0.951 0.514 0.927 0.897 0.912 0.870     5
##  6 0.550 0.844 0.805 0.857 0.829 0.793 0.853 0.652     4
##  7 1     0.956 0.854 1     0.756 0.931 1     0.739     3
##  8 1     0.978 0.878 1     0.976 0.931 1     0.913     3
##  9 0.983 0.978 0.829 1     0.951 0.966 0.882 0.783     2
## 10 1     0.978 0.829 1     1     0.966 1     0.870     4
## # ... with 340 more rows
```

Check balance

```
marks_CV %>% count(fold)
```

```
## # A tibble: 5 x 2
##    fold     n
##   <int> <int>
## 1     1    70
## 2     2    70
## 3     3    70
## 4     4    70
## 5     5    70
```

Fit models and get RMSE

```
RMSE_M1 <- double(5)
RMSE_M2 <- double(5)
```

```
for(i in 1:5){
  # Get test and train
  train <- marks_CV %>% filter(fold != i)
  test <- marks_CV %>% filter(fold == i)
  # Fit model to train
  M1 <- lm(E ~ OQ + A3 + A4 + A6, data = train)
  M2 <- lm(E ~ OQ + A3 + A6, data = train)
  # Predict for test
  M1_pred <- predict(M1, newdata = test)
  M2_pred <- predict(M2, newdata = test)
  # Calculate RMSE
  RMSE_M1[i] <- sqrt(mean((test$E - M1_pred)^2))
  RMSE_M2[i] <- sqrt(mean((test$E - M2_pred)^2))
}
```

```
mean(RMSE_M1)
```

```
## [1] 0.1738279
```

```
mean(RMSE_M2)
```

```
## [1] 0.1741081
```

So M2 has lower CV RMSE, so I will use this.

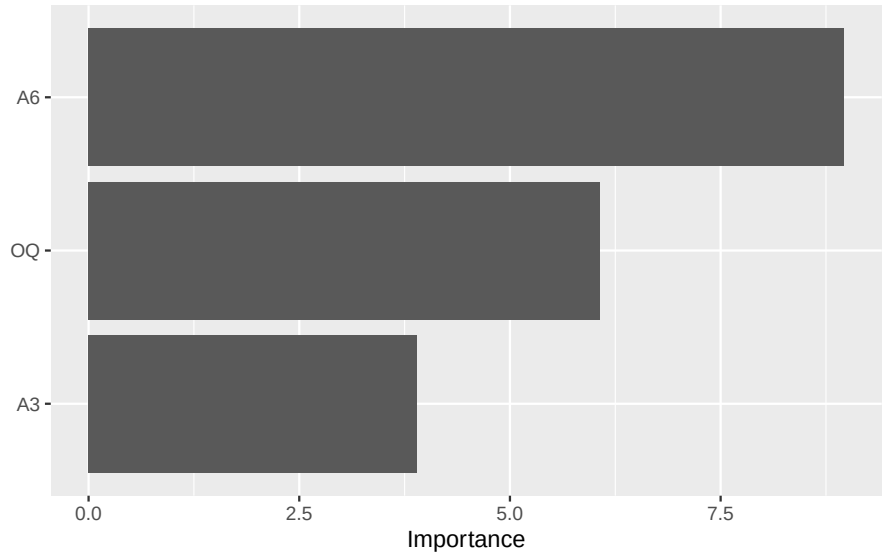
So final model is

```
Model 2: E ~ OQ + A3 + A6
```

```
final_model <- lm(E ~ OQ + A3 + A6, data = marks)
summary(final_model)
```

```
##
## Call:
## lm(formula = E ~ OQ + A3 + A6, data = marks)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.82442 -0.06864  0.02412  0.09377  0.63288
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.10625    0.02803   3.791 0.000177 ***
## OQ           0.21881    0.03608   6.064 3.47e-09 ***
## A3           0.16017    0.04108   3.899 0.000116 ***
## A6           0.36040    0.04021   8.964  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.1748 on 346 degrees of freedom
## Multiple R-squared:  0.574,  Adjusted R-squared:  0.5703
## F-statistic: 155.4 on 3 and 346 DF,  p-value: < 2.2e-16
```

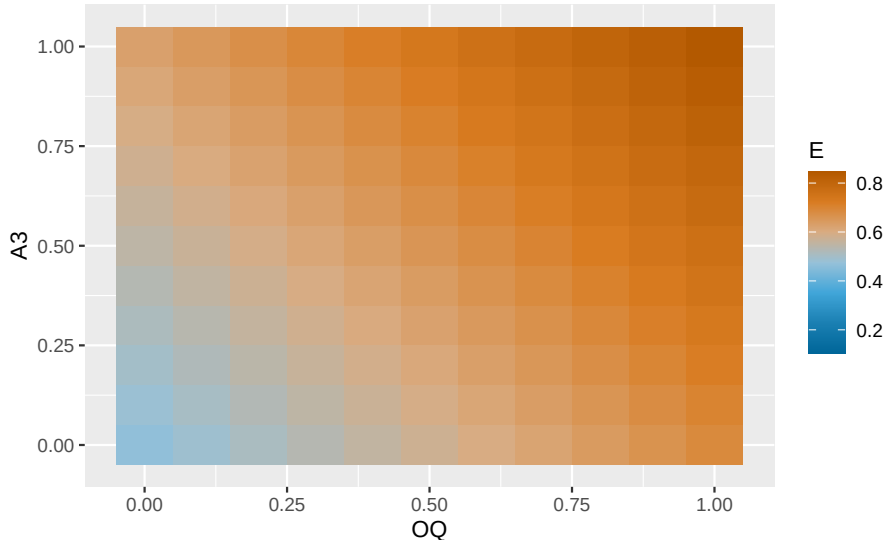
```
vip::vip(final_model)
```



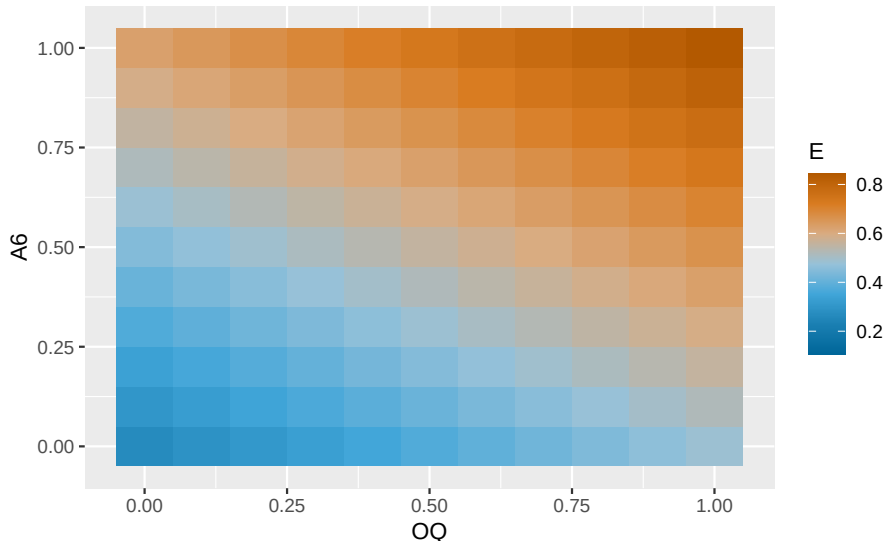

```
pred_grid <- crossing(
  OQ = seq(0, 1, 0.1),
  A3 = seq(0, 1, 0.1),
  A6 = seq(0, 1, 0.1)
)
pred_grid <-
  pred_grid %>%
  add_column(
    E = predict(final_model, newdata = pred_grid)
  )
pred_grid
```

```
## # A tibble: 1,331 x 4
##       OQ    A3    A6     E
##    <dbl> <dbl> <dbl> <dbl>
##  1     0     0   0   0.106
##  2     0     0   0.1 0.142
##  3     0     0   0.2 0.178
##  4     0     0   0.3 0.214
##  5     0     0   0.4 0.250
##  6     0     0   0.5 0.286
##  7     0     0   0.6 0.322
##  8     0     0   0.7 0.359
##  9     0     0   0.8 0.395
## 10     0     0   0.9 0.431
## # ... with 1,321 more rows
```

```
pred_grid %>%
  ggplot(aes(OQ, A3, fill = E)) +
  geom_tile() +
  harrypotter::scale_fill_hp("Ravenclaw")
```




```
pred_grid %>%
  ggplot(aes(OQ, A6, fill = E)) +
  geom_tile() +
  harrypotter::scale_fill_hp("Ravenclaw")
```




```
pred_grid %>%
  ggplot(aes(A3, A6, fill = E)) +
  geom_tile() +
  harrypotter::scale_fill_hp("Ravenclaw")
```