error count and jadud regression

XXX

2023-07-08

Error count

use error count to predict midterm 1

```
# predict midterm1 with both compiler and runtime errors
summary(rfit(Midterm1 ~
             R_totalError_HW3+R_totalError_HW4+
             C_totalError_HW3+C_totalError_HW4,
           data = allError, scores = bentscores3))
```

```
## Call:
## rfit.default(formula = Midterm1 ~ R_totalError_HW3 + R_totalError_HW4 +
##     C_totalError_HW3 + C_totalError_HW4, data = allError, scores = bentscores3)
##
## Coefficients:
##                    Estimate Std. Error t.value   p.value
## (Intercept)      94.534848   2.157611 43.8146 < 2.2e-16 ***
## R_totalError_HW3 -0.137248   0.138343 -0.9921  0.322313
## R_totalError_HW4 -0.254083   0.152965 -1.6611  0.098216 .
## C_totalError_HW3 -0.055061   0.017845 -3.0856  0.002309 **
## C_totalError_HW4 -0.096316   0.046704 -2.0622  0.040433 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Multiple R-squared (Robust): 0.09644761
## Reduction in Dispersion Test: 5.52393 p-value: 3e-04
```

```
# predict midterm1 with only compiler errors
summary(rfit(Midterm1 ~
             C_totalError_HW3+C_totalError_HW4,
           data = allError, scores = bentscores3))
```

```
## Call:
## rfit.default(formula = Midterm1 ~ C_totalError_HW3 + C_totalError_HW4,
##     data = allError, scores = bentscores3)
##
## Coefficients:
##                    Estimate Std. Error t.value   p.value
## (Intercept)      92.080364   1.801472 51.1139 < 2.2e-16 ***
## C_totalError_HW3 -0.060857   0.017913 -3.3974 0.0008145 ***
## C_totalError_HW4 -0.096988   0.047204 -2.0547 0.0411558 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
##
## Multiple R-squared (Robust): 0.07646293
## Reduction in Dispersion Test: 8.65193 p-value: 0.00025
```

use error count to predict midterm 2

```
# predict midterm2 with both compiler and runtime errors
summary(rfit(Midterm2 ~
             R_totalError_HW3+R_totalError_HW4 +
             R_totalError_HW5+R_totalError_HW6+
             R_totalError_HW7+R_totalError_HW8+
             C_totalError_HW3+C_totalError_HW4+
             C_totalError_HW5+C_totalError_HW6+
             C_totalError_HW7++C_totalError_HW8,
           data = allError,scores = bentscores3))
```

```
## Call:
## rfit.default(formula = Midterm2 ~ R_totalError_HW3 + R_totalError_HW4 +
##     R_totalError_HW5 + R_totalError_HW6 + R_totalError_HW7 +
##     R_totalError_HW8 + C_totalError_HW3 + C_totalError_HW4 +
##     C_totalError_HW5 + C_totalError_HW6 + C_totalError_HW7 +
##     +C_totalError_HW8, data = allError, scores = bentscores3)
##
## Coefficients:
##                      Estimate  Std. Error t.value   p.value
## (Intercept)       1.0199e+02  3.5425e+00 28.7908 < 2.2e-16 ***
## R_totalError_HW3 -1.3634e-01  1.5656e-01 -0.8708  0.385441
## R_totalError_HW4 -2.6788e-01  1.6388e-01 -1.6346  0.104539
## R_totalError_HW5 -6.2558e-02  1.5321e-02 -4.0831 7.688e-05 ***
## R_totalError_HW6 -1.2391e-02  1.3381e-02 -0.9260  0.356159
## R_totalError_HW7 -1.7985e-03  2.9560e-02 -0.0608  0.951578
## R_totalError_HW8 -9.8046e-02  3.3465e-02 -2.9298  0.004002 **
## C_totalError_HW3  1.0612e-02  2.1290e-02  0.4985  0.618999
## C_totalError_HW4 -8.8260e-02  5.4451e-02 -1.6209  0.107441
## C_totalError_HW5 -3.3935e-04  1.2279e-02 -0.0276  0.977994
## C_totalError_HW6 -2.1630e-02  2.5203e-02 -0.8582  0.392326
## C_totalError_HW7 -1.8009e-02  4.0593e-02 -0.4436  0.658031
## C_totalError_HW8 -4.7268e-02  2.3090e-02 -2.0471  0.042650 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Multiple R-squared (Robust): 0.2239622
## Reduction in Dispersion Test: 3.15052 p-value: 0.00057
```

```
# predict midterm2 with compiler errors
summary(rfit(Midterm2 ~
             C_totalError_HW3+C_totalError_HW4+
             C_totalError_HW5+C_totalError_HW6+
             C_totalError_HW7++C_totalError_HW8,
           data = allError,scores = bentscores3))
```

```
## Call:
## rfit.default(formula = Midterm2 ~ C_totalError_HW3 + C_totalError_HW4 +
##     C_totalError_HW5 + C_totalError_HW6 + C_totalError_HW7 +
##     +C_totalError_HW8, data = allError, scores = bentscores3)
```

```
##
## Coefficients:
##                     Estimate Std. Error t.value p.value
## (Intercept)      91.7130201  3.5674395 25.7084 < 2e-16 ***
## C_totalError_HW3  0.0024174  0.0223515  0.1082 0.91403
## C_totalError_HW4 -0.1441793  0.0577854 -2.4951 0.01378 *
## C_totalError_HW5 -0.0080038  0.0133583 -0.5992 0.55005
## C_totalError_HW6 -0.0240221  0.0276233 -0.8696 0.38602
## C_totalError_HW7 -0.0203218  0.0438426 -0.4635 0.64373
## C_totalError_HW8 -0.0516098  0.0252417 -2.0446 0.04281 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Multiple R-squared (Robust): 0.07921169
## Reduction in Dispersion Test: 1.96426 p-value: 0.07492
```

Jadud

use jadud to predict midterm 1 grades

```
# predict midterm1 with both compiler and runtime jadud measures
summary(rfit(Midterm1 ~ jadud_hw_03_c + jadud_hw_04_c + jadud_hw_03_r + jadud_hw_04_r, data = jadud_df))
```

```
## Call:
## rfit.default(formula = Midterm1 ~ jadud_hw_03_c + jadud_hw_04_c +
##     jadud_hw_03_r + jadud_hw_04_r, data = jadud_df)
##
## Coefficients:
##               Estimate Std. Error t.value   p.value
## (Intercept)    99.3683     1.6027 62.0022 < 2.2e-16 ***
## jadud_hw_03_c -47.9519     6.5295 -7.3439 2.398e-12 ***
## jadud_hw_04_c  -4.5003     5.8821 -0.7651    0.4449
## jadud_hw_03_r  13.5565    62.2608  0.2177    0.8278
## jadud_hw_04_r  -7.2937     8.9085 -0.8187    0.4137
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Multiple R-squared (Robust): 0.1847079
## Reduction in Dispersion Test: 15.46232 p-value: 0
```

```
# predict midterm1 with only compiler jadud measures
summary(rfit(Midterm1 ~ jadud_hw_03_c + jadud_hw_04_c, data = jadud_df))
```

```
## Call:
## rfit.default(formula = Midterm1 ~ jadud_hw_03_c + jadud_hw_04_c,
##     data = jadud_df)
##
## Coefficients:
##               Estimate Std. Error t.value   p.value
## (Intercept)    98.9173     1.6103 61.4297 < 2.2e-16 ***
## jadud_hw_03_c -49.2191     6.4254 -7.6601 3.186e-13 ***
## jadud_hw_04_c  -3.6324     5.7801 -0.6284    0.5302
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Multiple R-squared (Robust): 0.1810045
## Reduction in Dispersion Test: 30.38859 p-value: 0
```

3.2 use jadud to predict midterm 2 grades

```
# predict midterm2 with both compiler and runtime jadud measures
summary(rfit(Midterm2 ~ jadud_hw_03_c + jadud_hw_04_c + jadud_hw_05_c + jadud_hw_06_c + jadud_hw_07_c +
```

```
## Call:
## rfit.default(formula = Midterm2 ~ jadud_hw_03_c + jadud_hw_04_c +
##     jadud_hw_05_c + jadud_hw_06_c + jadud_hw_07_c + jadud_hw_08_c +
##     jadud_hw_03_r + jadud_hw_04_r + jadud_hw_05_r + jadud_hw_06_r +
##     jadud_hw_07_r + jadud_hw_08_r, data = jadud_df)
##
## Coefficients:
##               Estimate Std. Error t.value   p.value
```

```
## (Intercept)   103.2925     2.0754 49.7695 < 2.2e-16 ***
## jadud_hw_03_c -33.8606     7.5561 -4.4812 1.113e-05 ***
## jadud_hw_04_c   1.3468     6.3772  0.2112   0.83291
## jadud_hw_05_c   1.0471    11.2094  0.0934   0.92565
## jadud_hw_06_c -41.3119    17.4119 -2.3726   0.01839 *
## jadud_hw_07_c -16.9318     8.7138 -1.9431   0.05308 .
## jadud_hw_08_c -12.6171    10.9961 -1.1474   0.25227
## jadud_hw_03_r   2.3828    63.7165  0.0374   0.97020
## jadud_hw_04_r -10.7597     9.2594 -1.1620   0.24629
## jadud_hw_05_r -29.8552    16.6065 -1.7978   0.07337 .
## jadud_hw_06_r -37.8256    21.5767 -1.7531   0.08077 .
## jadud_hw_07_r  -5.4219     9.6043 -0.5645   0.57288
## jadud_hw_08_r -36.3899     8.2510 -4.4104 1.512e-05 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Multiple R-squared (Robust): 0.2635331
## Reduction in Dispersion Test: 7.75308 p-value: 0
```

```
# predict midterm2 with only compiler jadud measures
summary(rfit(Midterm2 ~ jadud_hw_03_c + jadud_hw_04_c + jadud_hw_05_c + jadud_hw_06_c + jadud_hw_07_c +
```

```
## Call:
## rfit.default(formula = Midterm2 ~ jadud_hw_03_c + jadud_hw_04_c +
##     jadud_hw_05_c + jadud_hw_06_c + jadud_hw_07_c + jadud_hw_08_c,
##     data = jadud_df)
##
## Coefficients:
##               Estimate Std. Error t.value   p.value
## (Intercept)    97.4574     1.3090 74.4538 < 2.2e-16 ***
## jadud_hw_03_c -39.9251     7.4531 -5.3568 1.834e-07 ***
## jadud_hw_04_c   4.3990     6.1913  0.7105   0.47801
## jadud_hw_05_c  -5.9471    11.0519 -0.5381   0.59096
## jadud_hw_06_c -40.9062    17.1230 -2.3890   0.01759 *
## jadud_hw_07_c -15.6508     8.6737 -1.8044   0.07230 .
## jadud_hw_08_c  -9.0145    10.9670 -0.8220   0.41183
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Multiple R-squared (Robust): 0.1900261
## Reduction in Dispersion Test: 10.40094 p-value: 0
```

HW grades

use HW grades to predict midterm 1

```
summary(rfit(Midterm1 ~ HW3 + HW4, data = grade, scores = bentscores3))
```

```
## Call:
## rfit.default(formula = Midterm1 ~ HW3 + HW4, data = grade, scores = bentscores3)
##
## Coefficients:
##             Estimate Std. Error t.value   p.value
## (Intercept) 48.723818   6.345296  7.6787 2.933e-13 ***
```

```
## HW3          0.436165   0.043363 10.0584 < 2.2e-16 ***
## HW4          0.026712   0.079818  0.3347    0.7381
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Multiple R-squared (Robust): 0.178964
## Reduction in Dispersion Test: 29.53539 p-value: 0
```

use HW grades to predict midterm 2

```
## Call:
## rfit.default(formula = Midterm2 ~ HW3 + HW4 + HW5 + HW6 + HW7 +
##     HW8, data = grade, scores = bentscores3)
##
## Coefficients:
##              Estimate Std. Error t.value   p.value
## (Intercept) 75.982253   6.742895 11.2685 < 2.2e-16 ***
## HW3          0.334412   0.055084  6.0710 4.348e-09 ***
## HW4         -0.024196   0.094596 -0.2558 0.7983117
## HW5         -0.497066   0.097770 -5.0840 6.961e-07 ***
## HW6          0.165118   0.048183  3.4269 0.0007069 ***
## HW7          0.137427   0.051657  2.6604 0.0082773 **
## HW8          0.074503   0.045673  1.6312 0.1040201
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Multiple R-squared (Robust): 0.2494467
## Reduction in Dispersion Test: 14.78959 p-value: 0
```