Sean Zhou

Appendices

A.1 Analysis of The First Fraction

WORKSHEET 1

Factorial Regression: Range versus Pull-back angle, Release angle, Eye hook position, Tension arm

Coded Coefficients

Term	Effect	Coef	SE Coef	T-Value	P-Value	VIF
Constant		51.88	*	*	*	
Pull-back angle	27.25	13.63	*	*	*	1.00
Release angle	-17.250	-8.625	*	*	*	1.00
Eye hook position	-10.750	-5.375	*	*	*	1.00
Tension arm	-19.250	-9.625	*	*	*	1.00
Pull-back angle*Release angle	-7.750	-3.875	*	*	*	1.00
Pull-back angle*Eye hook position	5.750	2.875	*	*	*	1.00
Pull-back angle*Tension arm	2.250	1.125	*	*	*	1.00

Analysis of Variance

Source	DF	Adj SS	Adj MS	F-Value	P-Value
Model	7	3248.87	464.12	*	*
Linear	4	3052.50	763.13	*	*
Pull-back angle	1	1485.12	1485.12	*	*
Release angle	1	595.13	595.13	*	*
Eye hook position	1	231.12	231.12	*	*
Tension arm	1	741.13	741.13	*	*
2-Way Interactions	3	196.38	65.46	*	*
Pull-back angle*Release angle	1	120.13	120.13	*	*
Pull-back angle*Eye hook position	1	66.13	66.13	*	*
Pull-back angle*Tension arm	1	10.12	10.12	*	*
Error	0	*	*		
Total	7	3248.87			

Regression Equation in Uncoded Units

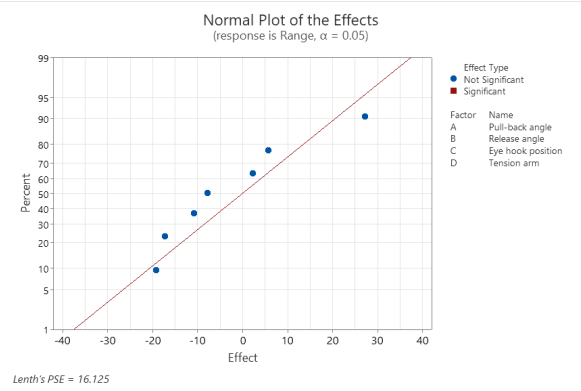
Range = 17.00 + 0.7583 Pull-back angle + 34.00 Release angle − 37.00 Eye hook position
− 22.00 Tension arm − 0.2583 Pull-back angle*Release angle
+ 0.1917 Pull-back angle*Eye hook position + 0.07500 Pull-back angle*Tension arm

Alias Structure

Factor	Name
A	Pull-back angle
B	Release angle
C	Eye hook position
D	Tension arm

Aliases

I + ABCD
A + BCD
B + ACD
C + ABD
D + ABC
AB + CD
AC + BD
AD + BC



Sean Zhou

A.2 Analysis of The Full Model

WORKSHEET 7

Factorial Regression: Range versus Pull-back angle, Release angle, Eye hook position, Tension arm

* NOTE * This design is not orthogonal.

The following terms cannot be estimated and were removed:
Pull-back angle*Release angle*Tension arm, Release angle*Eye hook position*Tension arm, Pull-back angle*Release angle*Eye hook position*Tension arm

Coded Coefficients

Term	Effect	Coef	SE Coef	T-Value	P-Value
Constant		50.60	2.36	21.43	0.000
Pull-back angle	29.21	14.60	2.36	6.19	0.000
Release angle	-11.29	-5.65	2.15	-2.63	0.034
Eye hook position	-11.54	-5.77	2.36	-2.44	0.044
Tension arm	-26.00	-13.00	2.59	-5.03	0.002
Pull-back angle*Release angle	-7.04	-3.52	2.15	-1.64	0.145
Pull-back angle*Eye hook position	-2.04	-1.02	3.36	-0.30	0.770
Pull-back angle*Tension arm	-11.25	-5.63	3.52	-1.60	0.155
Release angle*Eye hook position	10.21	5.10	3.50	1.46	0.188
Release angle*Tension arm	4.50	2.25	3.39	0.66	0.528
Eye hook position*Tension arm	-3.25	-1.63	2.59	-0.63	0.550
Pull-back angle*Release angle*Eye hook position	5.96	2.98	2.15	1.39	0.208
Pull-back angle*Eye hook position*Tension arm	-4.00	-2.00	2.59	-0.77	0.465

Term	VIF
Constant	
Pull-back angle	1.80
Release angle	1.50
Eye hook position	1.82
Tension arm	2.17
Pull-back angle*Release angle	1.51
Pull-back angle*Eye hook position	3.66
Pull-back angle*Tension arm	4.06
Release angle*Eye hook position	3.97
Release angle*Tension arm	3.60
Eye hook position*Tension arm	1.64
Pull-back angle*Release angle*Eye hook position	1.51
Pull-back angle*Eye hook position*Tension arm	2.19

Model Summary

S	R-sq	R-sq(adj)	R-sq(pred)
7.82091	96.75%	91.18%	*

Analysis of Variance

Source	DF	Adj SS	Adj MS	F-Value	P-Value
Model	12	12746.8	1062.23	17.37	0.000
Linear	4	10247.4	2561.84	41.88	0.000
Pull-back angle	1	2340.0	2340.00	38.26	0.000
Release angle	1	422.1	422.07	6.90	0.034
Eye hook position	1	365.4	365.38	5.97	0.044
Tension arm	1	1545.1	1545.14	25.26	0.002
2-Way Interactions	6	1691.5	281.91	4.61	0.033
Pull-back angle*Release angle	1	164.1	164.14	2.68	0.145
Pull-back angle*Eye hook position	1	5.6	5.64	0.09	0.770
Pull-back angle*Tension arm	1	155.8	155.77	2.55	0.155
Release angle*Eye hook position	1	129.9	129.92	2.12	0.188
Release angle*Tension arm	1	27.0	27.00	0.44	0.528
Eye hook position*Tension arm	1	24.1	24.14	0.39	0.550
3-Way Interactions	2	135.6	67.78	1.11	0.382
Pull-back angle*Release angle*Eye hook position	1	117.5	117.52	1.92	0.208
Pull-back angle*Eye hook position*Tension arm	1	36.6	36.57	0.60	0.465
Error	7	428.2	61.17		
Total	19	13174.9			

Regression Equation in Uncoded Units

Range = -637 + 5.35 Pull-back angle + 165 Release angle + 55 Eye hook position
- 62 Tension arm - 1.228 Pull-back angle*Release angle
- 0.463 Pull-back angle*Eye hook position + 0.292 Pull-back angle*Tension arm
- 27.7 Release angle*Eye hook position + 2.25 Release angle*Tension arm
+ 20.4 Eye hook position*Tension arm
+ 0.199 Pull-back angle*Release angle*Eye hook position
- 0.133 Pull-back angle*Eye hook position*Tension arm

Fits and Diagnostics for Unusual Observations

Sean Zhou

Obs	Range	Fit	Resid	Std Resid
5	31.00	31.00	−0.00	* X
7	84.00	84.00	0.00	* X
9	36.00	36.00	0.00	* X
11	72.00	72.00	−0.00	* X
14	42.00	42.00	−0.00	* X
15	35.00	35.00	0.00	* X
19	42.00	42.00	0.00	* X

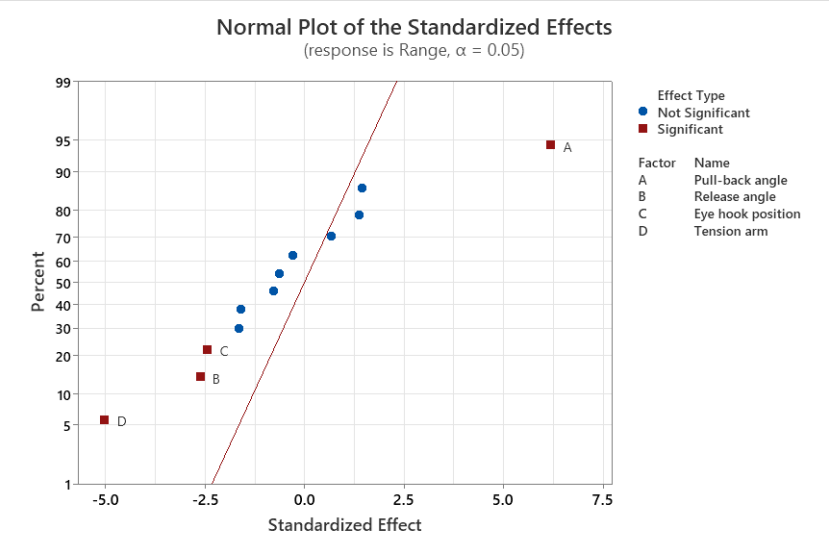
X Unusual X

Alias Structure

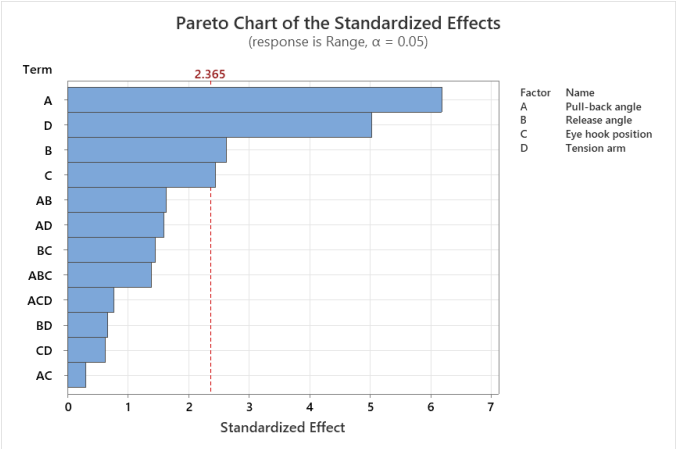
Factor	Name
A	Pull-back angle
B	Release angle
C	Eye hook position
D	Tension arm

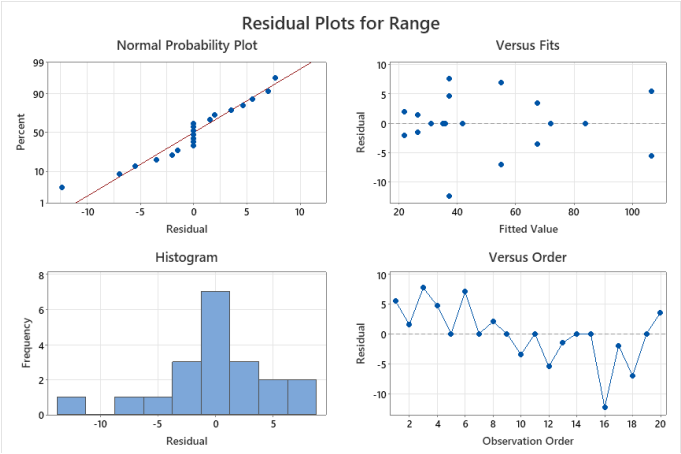
Aliases

I + ABCD
A + BCD
B + BCD
C + ABD
D − ABD
AB + ABCD
AC − ABD + BCD + ABCD
AD + ABD − BCD − ABCD
BC − ABD + BCD + ABCD
BD + ABD − BCD − ABCD
CD − ABCD
ABC + ABD
ACD − BCD



Sean Zhou





A.3 Firing Table

Sean Zhou

1 Sean Zhou

for $\hat{f}(x)$

	Range	Pullback Ang	Release Ang	Eyehook Posi	Tension Arm	Cup Position	Variance	95% CI Low	95% CI High
0	30	168	3	6	4	1	88.8685838	17.5327876	42.0672124
1	32	168	3	6	4	1	89.8685838	17.5327876	42.0672124
2	34	141	5	4	2	1	169.681502	9.99044862	58.7095514
3	36	143	5	4	2	1	156.140849	13.0381371	59.0618629
4	38	145	5	4	2	1	143.729054	16.0644273	59.4355727
5	40	148	5	4	2	1	132.446117	20.5543256	60.0456744
6	42	150	5	4	2	1	122.292038	23.5064815	60.4935185
7	44	152	5	4	2	1	113.266817	26.4173807	60.9826194
8	46	155	5	4	2	1	105.370454	30.6853849	61.8146151
9	48	157	5	4	2	1	98.6029498	33.4470248	62.4529752
10	50	159	5	4	2	1	92.9643034	36.1231574	63.1768426
11	52	162	5	4	2	1	88.4545152	39.9327876	64.4672124
12	54	149	3	4	2	1	92.9534938	39.7635158	66.8031509
13	56	151	3	4	2	1	90.5788824	44.0681575	69.3651758
14	58	152	3	4	2	1	88.3887784	46.197138	70.6695287
15	60	171	5	4	2	1	81.7039435	49.1461023	70.5538977
16	62	174	5	4	2	1	82.8384459	51.3038889	73.4961111
17	64	176	5	4	2	1	85.1018065	52.5015509	75.6984491
18	66	178	5	4	2	1	88.4940252	53.5327876	78.0672124
19	68	178	5	4	2	1	88.4940252	53.5327876	78.0672124
20	70	178	5	4	2	1	88.4940252	53.5327876	78.0672124
21	72	178	5	4	2	1	88.4940252	53.5327876	78.0672124
22	74	178	5	4	2	1	88.4940252	53.5327876	78.0672124
23	76	178	5	4	2	1	88.4940252	53.5327876	78.0672124
24	78	178	5	4	2	1	88.4940252	53.5327876	78.0672124
25	80	178	5	4	2	1	88.4940252	53.5327876	78.0672124
26	82	178	5	4	2	1	88.4940252	53.5327876	78.0672124
27	84	180	3	6	2	1	122.311895	65.5064815	102.493519
28	86	180	3	6	2	1	122.311895	65.5064815	102.493519
29	88	180	3	6	2	1	122.311895	65.5064815	102.493519
30	90	172	3	4	2	1	79.3866578	82.5625933	102.97074
31	92	172	3	4	2	1	79.3866578	82.5625933	102.97074
32	94	173	3	4	2	1	80.5176887	84.0036728	104.962994
33	96	174	3	4	2	1	81.8332272	85.4165183	106.983482
34	98	175	3	4	2	1	83.3332731	86.8034451	109.029888
35	100	176	3	4	2	1	85.0178265	88.166689	111.099978

Sean Zhou

A.4 Full Dataset

RunOrder	CenterPt	Blocks	Pull-back angle	Release angle	Eye hook position	Tension arm	Range
1	1	1	180	3	4	2	112
2	1	1	150	5	6	4	28
3	1	1	180	5	6	4	45
4	1	1	180	5	6	4	42
5	1	1	150	5	4	4	31
6	1	1	150	3	4	2	62
7	1	1	180	3	6	2	84
8	1	1	150	3	6	4	24
9	1	1	150	5	6	2	36
10	1	1	180	5	4	2	64
11	1	1	180	3	4	4	72
12	1	1	180	3	4	2	101
13	1	1	150	5	6	4	25
14	1	1	180	5	4	4	42
15	1	1	180	3	6	4	35
16	1	1	180	5	6	4	25
17	1	1	150	3	6	4	20
18	1	1	150	3	4	2	48
19	1	1	150	5	4	2	42
20	1	1	180	5	4	2	71

A.5 Python Code for Firing Table Generation

```
# %%
import pandas as pd
from numpy.linalg import inv
import numpy as np

from sympy.solvers import solve
from sympy import Symbol
import warnings
warnings.filterwarnings("ignore")

# %%
df = pd.read_excel('Statapult data.xlsx')
df = df.loc[:, 'RunOrder': 'Range']#.sort_values(by=['Range'])

firing = pd.read_csv('Firing Table Template.csv')

#Encoding the variables
def encode(df):
    df['A'] = [-1 if i < 180 else 1 for i in df['Pull-back angle'].values ]
    df['B'] = [-1 if i < 5 else 1 for i in df['Release angle'].values ]
    df['C'] = [-1 if i < 6 else 1 for i in df['Eye hook position'].values ]
    df['D'] = [-1 if i < 4 else 1 for i in df['Tension arm'].values ]
    return df

# %%
X = pd.read_excel('design matrix.xlsx')
Y = df['Range']
betahat = inv(X.T.dot(X)).dot(X.T.dot(Y)).reshape(13,1)

def regression_coded(A, B, C, D):
    '''Regression Model (Coded)'''
    X0 = np.array([1, A, B, C, D, A*B, A*C, A*D, B*C, B*D, C*D, A*B*C,
A*C*D]).reshape(13,1)

    return betahat.T.dot(X0)[0][0]
```

```
# %%
index = 0
for r in firing['Range'].values:
      temp = df[(df['Range'] >= r-10) & (df['Range'] <= r+10)]
      temp = encode(temp)

      sigma_sq = 61.17

      for row in temp.index:
            B = temp.loc[row, 'B']
            C = temp.loc[row, 'C']
            D = temp.loc[row, 'D']

            x = Symbol('x')

            V_temp = 100000
            best_setting = []

            #Calculate ideal angle
            angle_coded = solve(50.6 + 14.6*x - 5.65*B -5.77*C - 13*D - 3.52*x*B -
1.02*x*C - 5.63*x*D +5.10*B*C \
                        +2.25*B*D - 1.63*C*D + 2.98*x*B*C -2*x*C*D - r, x)[0]
            angle_original = angle_coded * 15 + 165
            #print (int(angle_original))
            #print (round(angle_coded, 2))
            if angle_original < 140 or angle_original > 180: #if the angle is not
possible
                  #print (r)
                  continue

            X_0 = get_X0(angle_coded, B, C, D)
            V_yhat = sigma_sq* (1+X_0.T.dot(inv(X.T.dot(X))).dot(X_0))[0][0]
            #print (row, angle_original, V_yhat, regression_coded(angle_coded, B,
C, D))

            if V_yhat < V_temp:
                  V_temp = V_yhat
                  #L = regression_coded(angle_coded, B, C, D) - 4.303 *
np.sqrt(float(V_yhat))

                  best_setting = [r, round(angle_original), temp.loc[row, 'Release
angle'], temp.loc[row, 'Eye hook position'], temp.loc[row, 'Tension arm'], 'cup',
V_temp, 'L', 'U']
```

```
      if best_setting == []:
            print (r)
      else:
            firing.loc[index] = best_setting

      print (best_setting)

      index+=1

firing.to_csv('final firing.csv')
```