An 2-level Factorial Experiment on Coagulation Time and Taste of Steamed Egg Custard

Yike Wang

May 6, 2020

Summary: A 2^5 unreplicated factorial experiment was carried out to explore the effects of five factors consisted in the cooking procedure on two yields: coagulation time and taste of egg custard. Egg-water ratio, type of bowl and their interaction have significant effects on coagulation time. Egg-water ratio, use of salt, water temperature, type of bowl and interaction between water and bowl have significant effects on taste. To get better taste and use less time, we should use high egg-water ratio.

1 Introduction

Steamed egg custard is a traditional and homely Chinese dish found all over China. I really like this easy and delicious dish because it always reminds me of my parents and childhood. The cooking steps for it are quite simple: beat the eggs and slowly add water to create a more tender texture; regularly stir them to fully mixed; pour the mixture into a container and scum the bubbles; steam. You are also free to add additional flavor like salt and solid ingredients like scallion. However, overcooking will generate cellular bubbles and lose soft. So one common difficulty of this dish is to control the cooking time critically to get the extent of just fully firm for a smooth and silky taste. In real life, to avoid overcooking, it could be annoying to repeatedly check whether the egg custard is fully cooked or not. The problem is more obvious when we use microwave or pressure pot for it, which I have experienced many times. Additionally, I have found that homemade egg custard always tastes differently compared with those served in restaurants, where homemade one has more egg flavor and restaurant egg custard could melt in your mouth.

As we all know, there could always exist some differences in our cooking procedure, such as the ratio of egg and water, the use of salt, the temperature of added water, the coverage of plastic wrap, the material or shape of our containers and so on. So I wonder how would these factors affect the coagulation time as well as taste? Therefore, I carried out these five factors factorial experiment to explore their effects and then offer guidance on better controlling cooking time and making more tasty egg custard in our daily life.

2 Experimental Procedure

2.1 Factors and Factor Levels

In this 32 runs, 2^5 unreplicated factorial experiment, I explored five factors consisted in the cooking procedure to test their effects on two yields: coagulation time and taste of egg custard. The factors and factor levels are listed in Table1, which I have made some changes against the proposal.

Variables	**Factors**	**Low Level (-)**	**High Level (+)**
A	Egg-Water Ratio	One Egg + Water ($\approx 1:2$)	Two Eggs + Water ($\approx 2:1$)
B	Use of Salt	No	2g
C	Coverage of Plastic Wrap	No	Yes
D	Water Temperature	$20°C$	$40°C$
E	Container Type	Ceramics Bowl	Stainless Steel Bowl

Table 1: Factors and Levels

Further explanation of factors:

A. Control the total volume of the mixed liquid of water and egg in each run to be 6oz by mixing them in a measuring cup.

B. Use the food scale to measure 2g salt for high level and add no salt for low level.

C. Cover the bowl entirely with plastic wrap for high level and cover nothing for low level.

D. Use thermometer to control water temperature of $20°C$ and $40°C$ for low and high levels.

E. Use a stainless steel bowl for high level and a ceramics bowl, which is a little rounder, for low level. Because under COVID-19 situation, it is hard to find two bowls sharing the same suitable shape with different materials. So I would consider two different types of bowl as factor E instead of considering materials only.

2.2 Measurement of Response

The experiment was realized in an AROMA 3-in-1 Super Pot with a transparent lid as a steamer. Before each run, I added the water to the level of the steam rack and boiled it first to guarantee equal temperature and environment in the pot. During the cooking procedure, the temperature of the pot was held at $350°F$.

1) **Coagulation Time**

   Start timing at the time I put the bowl in and close the lid. To better identify and record the exact time of just getting fully firm, I always put two scallion pieces in the middle of the surface, and later I waggled the pot to see when the scallion pieces are stable with egg and record that time. When I was not certain, I paused the time, turned the temperature to "warm" and used the spoon to test whether the middle of the surface is coagulated, and started timing when the water boiled again. As the time record is somehow subjective, I use 15 seconds as a period.

2) **Taste**

   I invited two of my roommates to be the tasters of egg custard. As I found out in the experiment that the mouthfeel and taste of egg custard would change greatly with the pass of time, I could only let them taste immediately when the egg custard was cooked. Luckily, the tasters have eaten egg custard from childhood and also I always cooked steamed egg custard for them, so they have enough experience in tasting egg custard.

   Tasters would give out one score–the taste of egg. The score criterion is: based on your experience and preference, score the taste of egg custard. The more egg flavor it contains, the higher the taster's score is.

2.3 Blocks and Runs

I divided 32 runs to 4 days as blocks. To realize this, I used ABC and CDE to define blocks, and thus blocks confounded with ABC, CDE and ABDE effects. From April 18th to April 21st (denoted as Day1, Day2, Day3, Day4), 8 runs were carried out each afternoon.

To do randomization, I wrote 8 tickets with standard order and five factors with their levels and put them in a black box, then randomly pick tickets out one by one to be the new order. In the latter three days, I repeated this procedure.

The corresponding runs and order performed within each block are listed in Table2.

Block	**Levels of ABC, CDE**	**Selected Runs**	**Order**
Day1	(-,-)	(1),ab,acd,bcd,ace,bce,de,abde	2,7,5,1,4,6,3,8
Day2	(-,+)	ac,bc,d,abd,e,abe,acde,bcde	2,6,3,4,1,8,7,5
Day3	(+,-)	a,b,cd,abcd,ce,abce,ade,bde	6,8,4,3,1,2,5,7
Day4	(+,+)	c,abc,ad,bd,ae,be,cde,abcde	8,4,1,3,2,7,5,6

Table 2: Blocks and Selected Runs

2.4 Cost of Time and Money

1) **Cost of Time**

   For each runs, the average time costed is 20 minutes, including preparing, steaming, tasting and washing time. On the first day, I did first three unsuccessful runs and found out some technical problems and the tasting problem, which cost me another two hours. The total amount of time spent on the experiment is 14 hours.

2) **Cost of Money**

   The cost of money lied in buying 5 boxes of eggs, a measuring cup, a food scale, and the stainless steel containers, which were in total $2.95 \times 5 + 4.99 + 10.5 + 2.99 = \$33.23$

3 Presentation and Analysis of Data

3.1 Data Presentation

The experimental data would be presented with blocks and standard orders were assigned within each block. For the second yield– taste, three numbers under each taster represent mouthfeel and egg flavor separately.

The detailed data is systematically listed in Table3.

In the upcoming sections of analysis, I divided them into two parts by two response.

Block	Standard Order	Run Order	Run	Factor					Yield1: Time	Yield2: Taste	
				A	B	C	D	E		Taster1	Taster2
Day1	1	2	(1)	-	-	-	-	-	10.5	4	4
Day1	2	7	ab	+	+	-	-	-	10	7	6
Day1	3	5	acd	+	-	+	+	-	12.5	7	7
Day1	4	1	bcd	-	+	+	+	-	15	8	6
Day1	5	4	ace	+	-	+	-	+	10.25	6	5
Day1	6	6	bce	-	+	+	-	+	9.5	4	4
Day1	7	3	de	-	-	-	+	+	11.25	4	3
Day1	8	8	abde	+	+	-	+	+	7.25	7	6
Day2	1	2	ac	+	-	+	-	-	15.75	7	7
Day2	2	6	bc	-	+	+	-	-	10.75	4	4
Day2	3	3	d	-	-	-	+	-	16.75	6	6
Day2	4	4	abd	+	+	-	+	-	10	8	7
Day2	5	1	e	-	-	-	-	+	11.75	3	3
Day2	6	8	abe	+	+	-	-	+	7.75	3	3
Day2	7	7	acde	+	-	+	+	+	8.25	4	7
Day2	8	5	bcde	-	+	+	+	+	9.75	5	4
Day3	1	6	a	+	-	-	-	-	13	5	5
Day3	2	8	b	-	+	-	-	-	8.5	5	5
Day3	3	4	cd	-	-	-	+	+	16.5	4	5
Day3	4	3	abcd	+	+	+	+	-	15.5	8	8
Day3	5	1	ce	-	-	+	-	+	10	3	3
Day3	6	2	abce	+	+	+	-	+	7.75	7	6
Day3	7	5	ade	+	-	-	+	+	7	6	6
Day3	8	7	bde	-	+	-	+	+	9.5	6	5
Day4	1	8	c	-	-	+	-	-	17.75	3	4
Day4	2	4	abc	+	+	+	-	-	15.25	6	7
Day4	3	1	ad	+	-	-	+	-	11.75	7	8
Day4	4	3	bd	-	+	-	+	-	11	5	5
Day4	5	2	ae	+	-	-	-	+	7.25	6	6
Day4	6	7	be	-	+	-	-	+	11	6	4
Day4	7	5	cde	-	-	+	+	+	16.75	4	3
Day4	8	6	abcde	+	+	+	+	+	8	7	7

Table 3: Data Presentation

3.2 Analysis of Coagulation Time Data

3.2.1 Modeling and Goodness of Fit Checking

For this unreplicated 2^5 factorial experiment, I fitted the full model at the beginning.

$$\begin{aligned}
y =& \eta + \frac{A}{2}x_1 + \frac{B}{2}x_2 + \frac{C}{2}x_3 + \frac{D}{2}x_4 + \frac{E}{2}x_5 + \frac{AB}{2}x_1x_2 + \frac{AC}{2}x_1x_3 + \frac{AD}{2}x_1x_4 + \frac{AE}{2}x_1x_5 \\
&+ \frac{BC}{2}x_2x_3 + \frac{BD}{2}x_2x_4 + \frac{BE}{2}x_2x_5 + \frac{CD}{2}x_3x_4 + \frac{CE}{2}x_3x_5 + \frac{DE}{2}x_4x_5 \\
&+ \frac{ABD}{2}x_1x_2x_4 + \frac{ABE}{2}x_1x_2x_5 + \frac{ACD}{2}x_1x_3x_4 + \frac{ACE}{2}x_1x_3x_5 \\
&+ \frac{ADE}{2}x_1x_4x_5 + \frac{BCD}{2}x_2x_3x_4 + \frac{BCE}{2}x_2x_3x_5 + \frac{BDE}{2}x_2x_4x_5 \\
&+ \frac{ABCD}{2}x_1x_2x_3x_4 + \frac{ABCE}{2}x_1x_2x_3x_5 + \frac{BCDE}{2}x_2x_3x_4x_5 + \frac{ABCDE}{2}x_1x_2x_3x_4x_5 \\
&+ \delta_1 z_1 + \delta_2 z_2 + \delta_3 z_3 + \epsilon
\end{aligned} \tag{1}$$

I first used Lenth's method an Dong's method to detect significant effects. Fit the full model and get estimates of effects in Table 4.

Effect	**Estimate**	**Effect**	**Estimate**	**Effect**	**Estimate**	**Effect**	**Estimate**
A	-1.84	BC	-0.16	DE	-0.28	ADE	0.56
B	-1.88	AD	-1.5	ABD	-0.81	BDE	-0.78
C	2.22	BD	0.09	ACD	-0.34	ABCD	0.03
D	0.59	CD	0.06	BCD	0.5	ABCE	-0.38
E	-3.56	AE	-1.41	ABE	-0.41	ACDE	-0.91
AB	1.41	BE	0.38	ACE	0.06	BCDE	-0.88
AC	0.25	CE	-1.28	BCE	-0.91	ABCDE	0.97

Table 4: Effect Estimates in Full Model

For Lenth's method, to get rid of outliers, we have $s_0 = 1.03$ and $PSE = 0.89$.

Here $g = 28 \quad v = 9.33 \quad \gamma = 0.00188$, and the the bound is $t_{v,\gamma} \times PSE = 3.41$. Only E is shown significant.

Then for Dong's method, by calculation I get $m_1 = m_2 = 27 \quad s_1^2 = s_2^2 = 0.96$, and the bound is $t_{m_2,\gamma} \times s_2 = 3.11$. Again, only E is shown significant.

Both of them suggest that only E effect is significant.

Check the goodness of fit test for the reduced model

$$y = \eta + \frac{E}{2}x_5 + \delta_1 z_1 + \delta_2 z_2 + \delta_3 z_3 + \epsilon \tag{2}$$

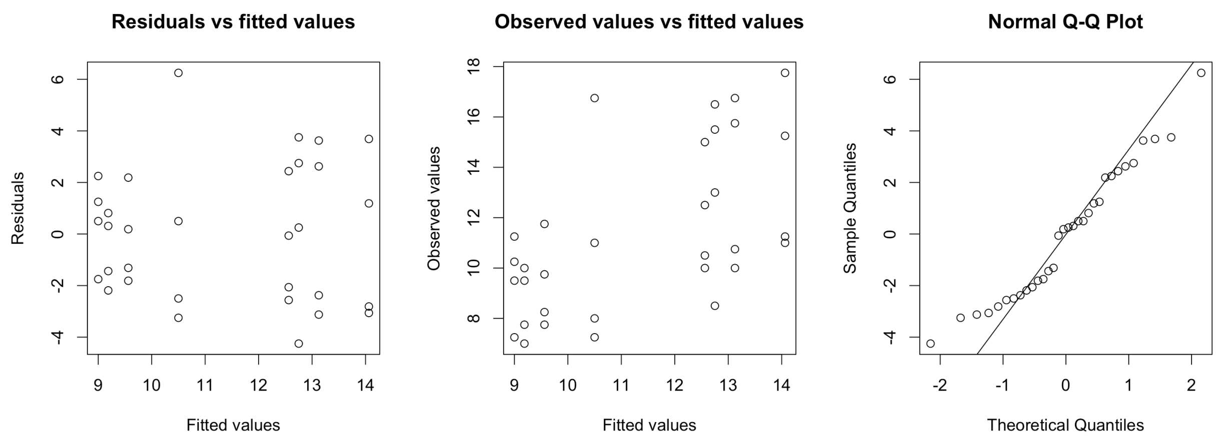

Figure 1: Residual plots of Model (2)

The first plot suggests constant variance in residuals since the residuals seem randomly distributed against fitted values and no relationship with fitted values. The second plot shows not a really strong association between observed values and fitted values and they have a relatively large dispersion, which may suggest the model is not good enough. But the third normal QQ plot fits not well and thus the normality assumption may be violated.

Although the former model implies that the interaction terms may be negligible, I could not make the strong assumption that there is no interaction directly. Instead, I tried to eliminate some high level interaction terms level by level, assuming them to be negligible to estimate $\hat{\sigma}^2$, and tried box-cox transformation when necessary.

eliminate	5fi	4fi	3fi	2fi
λ	<-2	<-2	-1.27	-0.46

Table 5: λ chose in each transformation

During eliminating process, the chosen of λ is given in Table5. When the 3fi and higher level interaction terms are assumed negligible and only main effects and 2fi remained, the transformation performs well and provides an improvement in normality, comparing Figure2 and Figure3.

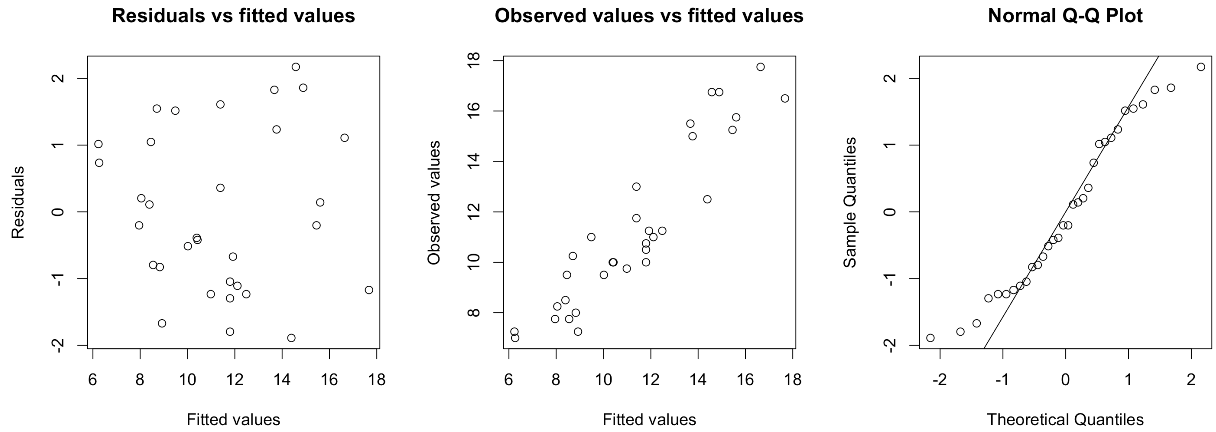

Figure 2: Residual plots of Model with only main effects and 2fi

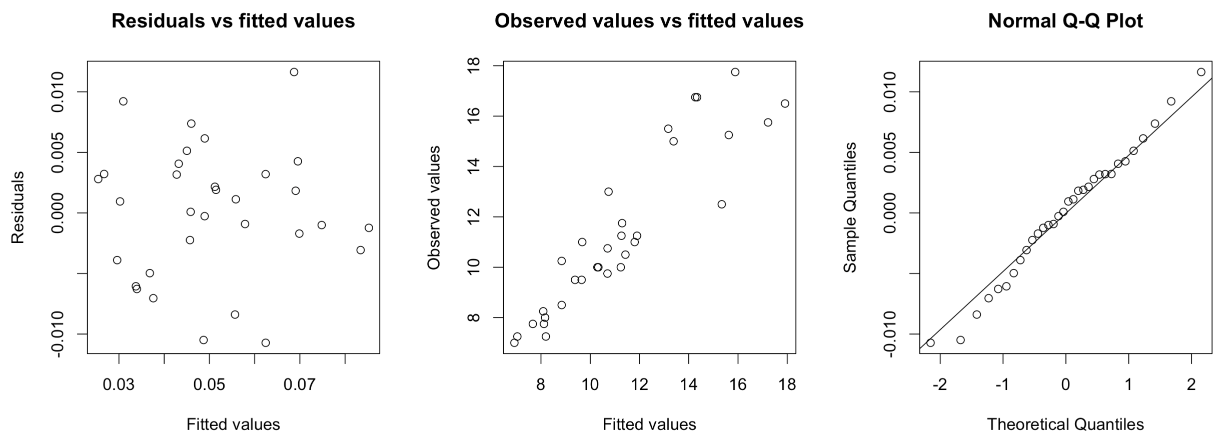

Figure 3: Residual plots after transformation

Therefore I kept the transformation model (3)

$$y^{-1.27} = \eta + \frac{A}{2}x_1 + \frac{B}{2}x_2 + \frac{C}{2}x_3 + \frac{D}{2}x_4 + \frac{E}{2}x_5 + \frac{AB}{2}x_1x_2 + \frac{AC}{2}x_1x_3 + \frac{AD}{2}x_1x_4 + \frac{AE}{2}x_1x_5 + \delta_1 z_1 + \delta_2 z_2 + \delta_3 z_3 + \epsilon \quad (3)$$

Compared with the former residual plots of the reduced model, the plots in transformation model perform much better. The first plot shows constant variance of residuals and no relationship with fitted values. The second plot shows a relatively strong relationship between observed and fitted data with much less dispersion. The normal QQ plot also fits much better with most points distributed very close to the quantile line. In conclusion, model (3) is chosen to be the final model.

Then I used simultaneous confidence intervals using studentized maximum modulus distribution to detect significant effects in the transformed model.

The estimates of effect in model (3) is shown in Table6.

Effect	Estimate	Effect	Estimate	Effect	Estimate
A	0.012	AB	-0.0049	BD	-0.0012
B	0.0085	AC	-0.0035	BE	-0.00096
C	-0.011	AD	-0.0070	CD	-0.00050
D	-0.0018	AE	0.012	CE	0.0038
E	0.0201	BC	0.00074	DE	0.0029

Table 6: Effect Estimates of Model(3)

In this model, $N = 32 \quad \hat{\sigma} = 0.008355 \quad v = 13 \quad g = 15 \quad M_{g,v}^{(0.1)} = 3.1$. Therefore, the bound becomes $M_{g,v}^{(0.1)}\sqrt{4\hat{\sigma}^2/N} = 0.0092$.

Compared this bound with the absolute value of effect estimates, A, C, E and AE are shown significant. Now we could find out that this transformation reveals more significant effects than using Lenth's and Dong's Method.

3.2.2 Time Dependence Checking

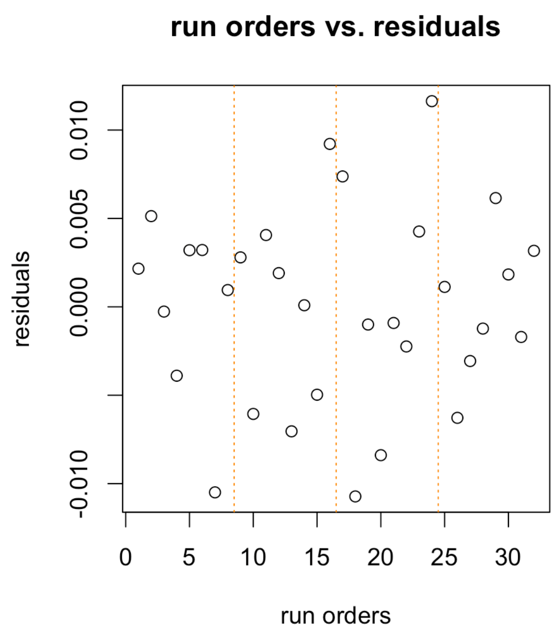

Figure 4: Residuals vs. Run orders

In Figure 4, I plotted residuals of the fitted model versus run order to check for time dependence. As the plot showed, residuals were randomly dispersed among run order, which implies the independence of time in this experiment.

3.2.3 Fitted Model and Fitted Values

In the choice of final model, I kept the insignificant terms remain in the model. Therefore, the fitted model for coagulation time I chose is

$$
\begin{aligned}
\hat{y}^{-1.27} =& 0.051 + 0.0061x_1 + 0.0043x_2 - 0.0053x_3 - 0.00092x_4 + 0.01x_5 \\
& - 0.0025x_1x_2 - 0.0018x_1x_3 + 0.0035x_1x_4 + 0.0062x_1x_5 + 0.00038x_2x_3 - 0.0006x_2x_4 \\
& - 0.00048x_2x_5 - 0.00025x_3x_4 + 0.0019x_3x_5 + 0.0014x_4x_5 - 0.0013z_1 + 0.0025z_2 - 0.0043z_3
\end{aligned} \tag{4}
$$

The fitted value for each set of experimental conditions is listed in Table7, where "Cond" represents for Experimental Condition.

Cond	**Fitted Value**	**Cond**	**Fitted Value**	**Cond**	**Fitted Value**	**Cond**	**Fitted Value**
(1)	11.43	d	14.34	e	11.29	de	11.90
a	10.74	ad	11.27	ae	8.20	ade	6.92
b	8.84	bd	11.80	be	9.67	bde	9.65
ab	10.33	abd	10.29	abe	7.67	abde	7.04
c	15.89	cd	17.91	ce	11.24	cde	14.26
ac	17.22	acd	15.34	ace	8.84	acde	8.09
bc	10.70	bcd	13.38	bce	9.38	bcde	10.70
abc	15.62	abcd	13.17	abce	8.12	abcde	8.17

Table 7: Experimental Conditions and corresponding Fitted Values

3.3 Analysis of Taste Data

3.3.1 Modeling and Goodness of Fit Checking

When I followed the same procedure for analyzing unreplicated data as above, I found out that two tasters share different preferences on egg custard taste, which suggested that two tasters could be regarded as a taster block and then the taster scores could be regarded as replicated experiments. Based on the length limit, I would omit the details for these analysis and only list my results in Table8 for further analyses.

	Taster1	Taster2
Final Model	Main Effect+2fi	Additive Model
Significant Effects	A,B	A,D,E

Table 8: Unreplicated Analysis for Taster1 and Taster2

Treat two tasters as a taster block with factor 1 and 2, and insert to the model. I fitted the full model, and used simultaneous confidence intervals to figure out significant effects.

Effect	Estimate	Effect	Estimate	Effect	Estimate	Effect	Estimate
A	2.13	BC	-0.063	DE	-0.56	ADE	-0.063
B	0.94	AD	-0.19	ABD	1.3×10^{-16}	BDE	-2.2×10^{-16}
C	-0.13	BD	-1.1×10^{-16}	ACD	-0.19	ABCD	-3.0×10^{-16}
D	0.81	CD	0.063	BCD	0.50	ABCE	0.44
E	-0.75	AE	0.13	ABE	-0.063	ACDE	0.31
AB	-0.19	BE	0.31	ACE	-0.13	BCDE	-0.38
AC	0.25	CE	-0.25	BCE	-0.19	ABCDE	0.13

Table 9: Effect Estimates in Full Model

In this model, $N = 64 \quad \hat{\sigma} = 0.008355 \quad v = 31 \quad g = 28 \quad M^{(0.1)}_{g,v} = 3.1$. Then the bound becomes $M^{(0.1)}_{g,v}\sqrt{4\hat{\sigma}^2/N} = 0.53$. Compared with the bound and the absolute value of effect estimates, the significant effects are A,B,D,E,DE.

To check the goodness of fit of the model, I plotted three plots in Figure5.

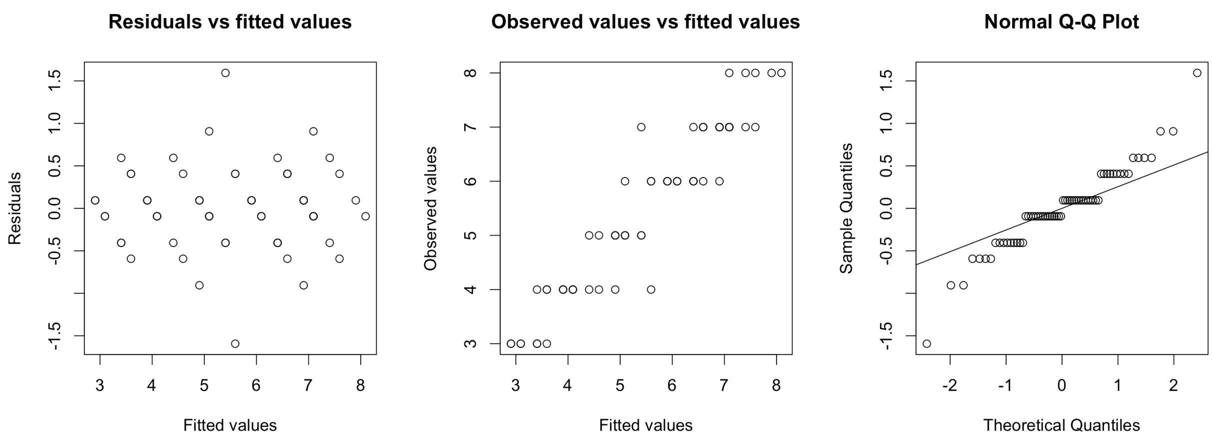

Figure 5: Residual plots of Full Model

It seems that the plot of residuals against fitted values shows a pattern of heteroscedasticity. But there are only two residuals with large values while other points show no pattern of unconstant variance. In addition, the Normal QQ plot does not perform well, which, in my opinion, is mostly due to the distribution of original data that scores mostly aggregate in several similar scores. To see whether these two points are outliers in the original data, I plot normal QQ plot of Y in Figure6.

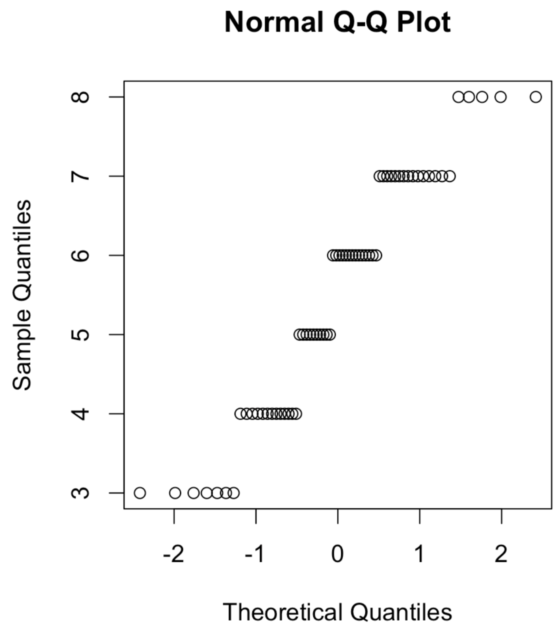

Figure 6: Normal QQ plot of taste scores

However, there is no outlier in original data. Then I tried transformation for this problem, whose results are shown in Figure7, but the assumptions still could not be satisfied and plots perform no better. Consequently, I kept the full model as the final model while it is not that reliable.

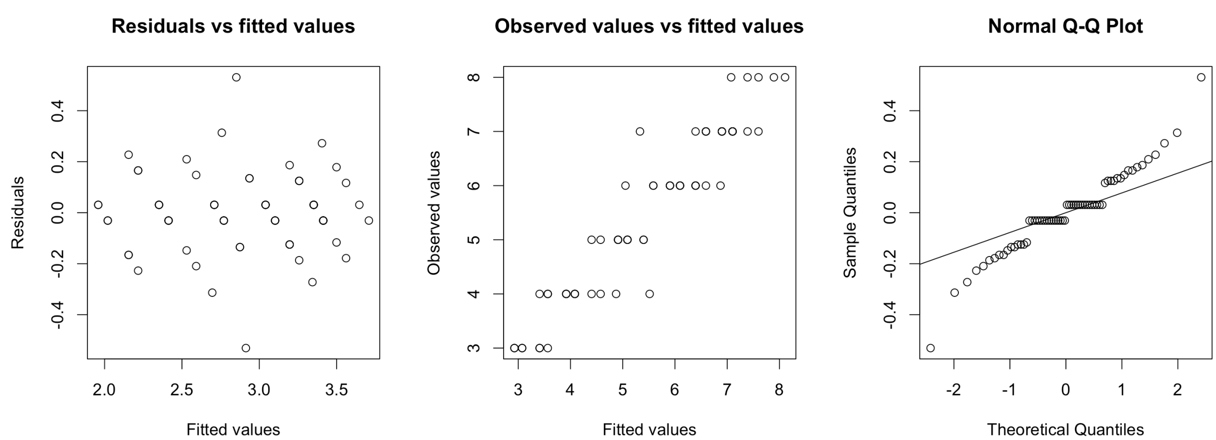

Figure 7: Residual plots of Transformation of Full Model

3.3.2 Time Dependence Checking

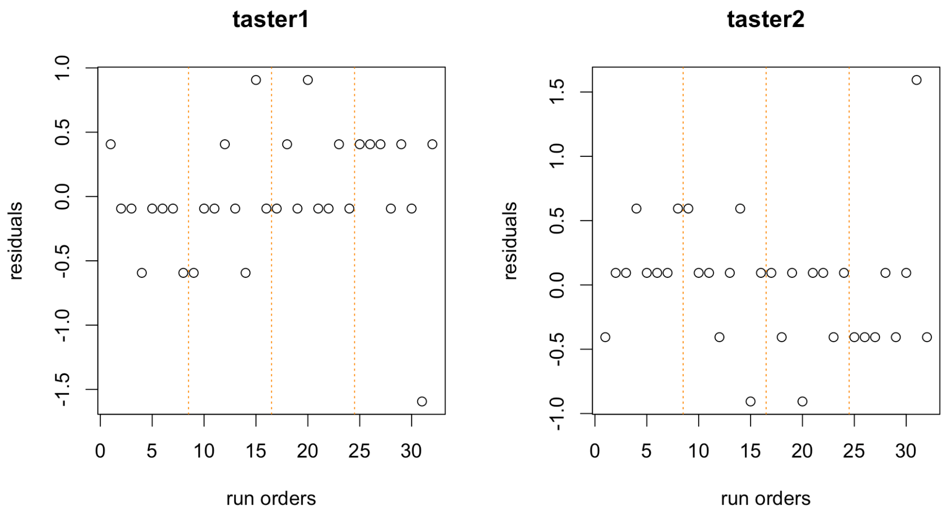

Figure 8: Run order vs. Residuals

To check time dependence, I separately plot two figures for data of two tasters in Figure8. As we can see, despite one outlier in these plots, the residuals show no relationship with run orders. Therefore, we could conclude the time independence.

3.3.3 Fitted Model and Fitted Values

$$
\begin{aligned}
\hat{y} =& 5.59 + 1.06x_1 + 0.47x_2 - 0.063x_3 + 0.41x_4 - 0.38x_5 - 0.094x_1x_2 + 0.13x_1x_3 - 0.094x_1x_4 \\
& 0.063x_1x_5 + 0.031x_2x_3 - 5.5 \times 10^{-17}x_2x_4 + 0.16x_2x_5 + 0.031x_3x_4 - 0.13x_3x_5 - 0.28x_4x_5 \\
& 6.5 \times 10^{-17}x_1x_2x_4 - 0.094x_1x_3x_4 + 0.25x_2x_3x_4 - 0.031x_1x_2x_5 - 0.063x_1x_3x_5 \\
& - 0.094x_2x_3x_5 - 0.031x_1x_4x_5 - 1.1 \times 10^{-16}x_2x_4x_5 \\
& - 1.5 \times 10^{-16}x_1x_2x_3x_4 + 0.22x_1x_2x_3x_5 + 0.16x_1x_3x_4x_5 - 0.19x_2x_3x_4x_5 + 0.063x_1x_2x_3x_4x_5 \\
& + 0.063z_1 - 0.063z_2 + 3.5 \times 10^{-16}z_3 - 0.19taster
\end{aligned}
\tag{5}
$$

Model (5) is the fitted model. Fitted values for taster1 and taster2 are shown separately in Table10 and Table11.

Cond	Fitted Value	Cond	Fitted Value	Cond	Fitted Value	Cond	Fitted Value
(1)	4.09	d	6.09	e	3.09	de	3.59
a	5.09	ad	7.59	ae	6.09	ade	6.09
b	5.09	bd	5.09	be	5.09	bde	5.59
ab	6.59	abd	7.59	abe	7.09	abde	6.59
c	3.59	cd	4.59	ce	3.09	cde	3.59
ac	7.09	acd	7.09	ace	5.59	acde	5.59
bc	4.09	bcd	7.09	bce	4.09	bcde	4.59
abc	6.59	abcd	8.09	abce	6.59	abcde	7.09

Table 10: Experimental Conditions and corresponding Fitted Values for Taster 1

Cond	Fitted Value	Cond	Fitted Value	Cond	Fitted Value	Cond	Fitted Value
(1)	3.91	d	5.91	e	2.91	de	3.41
a	4.91	ad	7.41	ae	5.91	ade	5.91
b	4.91	bd	4.91	be	4.91	bde	5.41
ab	6.41	abd	7.41	abe	6.91	abde	6.41
c	3.41	cd	4.41	ce	2.91	cde	3.41
ac	6.91	acd	6.91	ace	5.41	acde	5.41
bc	3.91	bcd	6.91	bce	3.91	bcde	4.41
abc	6.41	abcd	7.91	abce	6.41	abcde	6.91

Table 11: Experimental Conditions and corresponding Fitted Values for Taster 2

4 Conclusions from the Experiment

1) **Time Data.** As the fitted model showed, the egg-water ratio, coverage of plastic wrap, type of bowl and also the interaction between egg-water ratio and type of bowl have significant effect on the coagulation time of steamed egg custard.

   Since the fitted model has value $\lambda = -1.27$, the effects have opposite effect on the coagulation time against what their signs imply. For instance, the egg-water ratio has a positive sign, which means that the higher the ratio of egg and water is, the less time the coagulation of egg custard will take. As Table6 shown, A,E and AE have positive signs and C has negative signs. In conclusion, high ratio of egg and water (using two egg and water to be 6oz), using the second type of bowl with stainless steel material will reduce the cooking time. Moreover, if we make egg custard with the second type of bowl, then the time decrease in making high egg-water ratio instead of low egg-water ratio will be larger than using the first type of bowl, which is made of ceramics.

   To reach the minimum coagulation time when making steamed egg custard, one should use high egg-water ratio, use the second type of bowl and not cover the plastic wrap. However, minimizing time is always not our goal, taste instead, but we can get some inspiration from this design to control these factors to better adjust cooking time in our daily life.

2) **Taste Data.** As the fitted model showed, the egg-water ratio, use of salt, water temperature, type of bowl and also the interaction between water temperature and type of bowl have significant effect on the taste of steamed egg custard.

   As Table9 shown, A,B,D have positive signs while E and DE have negative signs. This implies that on the one hand, high egg-water ratio, second type of bowl and 40 degree warm water will make egg custard tastier. On the other hand, the usage of second type bowl will reduce the taste of egg custard. Additionally, when using the second type bowl, the increase in taste from using 20 degree cold water to 40 degree hot water will be less than that using the first type bowl.

   To reach the best taste, one should use high egg-water ratio, add salt, use 40 degree warm water and use type one bowl instead.

   Seen from the final models of two tasters, all their significant effects also have significant effect in the replicated model.

Comparing these two yields and their significant effects, we could know that using the high egg-water ratio, where two eggs are used, will not only reduce cooking time but also increase the taste.

5 Recommendations and Lessons Learned

Based on the results, I feel quite surprised about the significant effect of bowl type on taste. The possible reason behind I could think of is that the main influence given by these bowls on taste consists in the shape difference rather than materials. To check whether this thought is reasonable, future experiments are needed to control container material and test the effect of shape only on taste score.

In the taste score analysis, the problem remained is the outlier in residual plots. One possible reason may be the number of replication is only 2. But I could not figure out exactly why they show like this. But reflecting on my experiment process, there are some difficulties that may relate to the problem in taste scores. The first one is that even the tasters are adequately experienced, they could not make comparisons among runs due to the characteristic of egg custard, which may result in possible score error. The second problem lies in the difficulty of full coagulation in two runs and a small amount of water stratified in several runs with low egg-water ratio. These may confuse the judgement of tasters.

The second problem listed above could also influence the cooking time. At the beginning of the experiment, I confronted with this difficulty and failed in recording time for one hard to coagulation run. But the egg-water ratio is hard to change in a relatively small total volume. So I tried my best to adjust the experiment procedure to overcome it and use 15s as a recording period to judge and record. However, the time recording may still be not precise. For future experiments, the experimenter could perform some runs ahead and figure out possible difficulties and adjust the procedure in time.

6 Appendices

```
###########################cooking time

y1<-c(10.5,13,8.5,10,17.75,15.75,10.75,15.25,16.75,
       11.25,11,10,16.5,12.5,15,15.5,
       11.75,7.25,11,7.75,10,10.25,9.5,7.75,
       11.25,7,9.5,7.25,16.75,8.25,9.75,8)

A = rep(c(-1,1),16)
B = rep(c(-1,-1,1,1), 8)
C = rep(c(rep(-1,4),rep(1,4)), 4)
D = rep(c(rep(-1,8),rep(1,8)), 2)
E = c(rep(-1,16), rep(1,16))
Block<-c('1','3','3','1','4','2','2','4','2','4','4','2','3','1','1','3',
          '2','4','4','2','3','1','1','3','1','3','3','1','4','2','2','4')
time<-data.frame(y1,A,B,C,D,E,Block)
fit_time<-lm(y1~A*B*C*D*E+Block)
summary(fit_time)
anova(fit_time)

est.effects<-2*fit_time$coefficients[-c(1,7,8,9,20,29,32)]
a.est<-abs(est.effects)
g<-length(est.effects)

########Lenth
s0<-1.5*median(a.est)
a1<-a.est[a.est<2.5*s0]
pse<-1.5*median(a1)
v<-g/3
alpha<-0.1
ga<-0.5*(1-(1-alpha)^(1/g))
a.est[a.est>qt(ga,v,lower.tail = F)*pse]

#########Dong
m1<-length(a1)
s12<-sum(a1^2)/m1
s1<-sqrt(s12)
a2<-a.est[a.est<2.5*s1]
m2<-length(a2)
s22<-sum(a2^2)/m2
```

```
s2<-sqrt(s22)
a.est>s2*qt(ga,m2,lower.tail = F)
a.est[a.est>s2*qt(ga,m2,lower.tail = F)]

par(mfrow=c(1,3),pty="s",cex=1.2)
rd.1<-lm(y1~E+Block)#residual plots of dong&lenth method
summary(rd.1)
res1<-rd.1$residuals
fit1<-rd.1$fitted.values
plot(res1 ~ fit1,xlab="Fitted values",ylab="Residuals",main="Residuals vs fit
plot(y1 ~ fit1,xlab="Fitted values",ylab="Observed values",main="Observed val
qqnorm(res1)
qqline(res1)

f1<-lm(y1~A+B+C+D+E+Block)
summary(f1)
resi <- f1$residuals#residual plots of transformation
fitt<- f1$fitted.values
plot(resi ~ fitt,xlab="Fitted values",ylab="Residuals",main="Residuals vs fit
plot(y1~ fitt,xlab="Fitted values",ylab="Observed values",main="Observed valu
qqnorm(resi)
qqline(resi)
########transformation

#notes method
gm <- exp(mean(log(y1))) ## geometric mean
ssr <- NULL
lambda.seq <- seq(from=-2,to=3,length.out=20)
for(lambda in lambda.seq){
  if(lambda == 0){
    y <- gm*log(y1)
  } else {
    y <- (y1^lambda-1)/(lambda * gm^{lambda-1})
  }
  fit <- lm(y ~ A+B+C+D+E+Block)
  ssr <- c(ssr,sum(fit$resid^2))
}
plot(lambda.seq,ssr,type="b",xlab=expression(lambda),
        ylab=expression(S[lambda]),main = "Select lambda")
y<-y1^(-0.5)
y<-y1^ll
```

```
model<-lm(y~A+B+C+D+E+Block)
summary(model)
anova(model)

resid <- model$residuals#residual plots of transformation
fitted <- model$fitted.values
plot(resid ~ fitted,xlab="Fitted values",ylab="Residuals",main="Residuals vs
plot(y ~ fitted,xlab="Fitted values",ylab="Observed values",main="Observed va
qqnorm(resid)
qqline(resid)
#seems A,B,C,E


####selection
library(MASS)
time.t1<-lm(y1~A*B*C*D*E+Block-A:B:C:D:E)
time.bc1<-boxcox(time.t1,lambda = seq(-5,5-0.05))
time.bc1$x[which.max(time.bc1$y)]

time.t2<-lm(y1~(A+B+C+D+E)^3+Block)
time.bc2<-boxcox(time.t2)
time.bc2$x[which.max(time.bc2$y)]

time.t3<-lm(y1~(A+B+C+D+E)^2+Block)
resi<-time.t3$residuals
fitt<-time.t3$fitted.values
plot(resi ~ fitt,xlab="Fitted values",ylab="Residuals",main="Residuals vs fit
plot(y1 ~ fitt,xlab="Fitted values",ylab="Observed values",main="Observed val
qqnorm(resi)
qqline(resi)

time.bc3<-boxcox(time.t3)
l<-time.bc3$x[which.max(time.bc3$y)]
y.t<-y1^l
model.t<-lm(y.t~(A+B+C+D+E)^2+Block)
summary(model.t)
anova(model.t)
resid.t<- model.t$residuals#residual plots of transformation
fitted.t <- model.t$fitted.values
fit.t<-fitted.t^(1/l)
plot(resid.t ~ fitted.t,xlab="Fitted values",ylab="Residuals",main="Residuals
```

```
plot(y1 ~ fit.t,xlab="Fitted values",ylab="Observed values",main="Observed va
qqnorm(resid.t)
qqline(resid.t)



#tranformation studentized mm
coef.t<-model.t$coefficients[-c(1,7,8,9)]
N<-32
g.t<-15
v.t<-13
M<-3.1
se<-0.0014770
coef.t[abs(coef.t)>M*se]
#A,C,E,AE



######time dependence?

run1<-c(2,7,5,1,4,6,3,8)
run2<-c(2,6,3,4,1,8,7,5)
run3<-c(6,8,4,3,1,2,5,7)
run4<-c(8,4,1,3,2,7,5,6)
run<-c(run1,run2+8,run3+16,run4+24)
plot(run,resid.t,xlab = "run orders",ylab = "residuals",main = "run orders vs
abline(v=8.5,col="darkorange",lty=3)
abline(v=16.5,col="darkorange",lty=3)
abline(v=24.5,col="darkorange",lty=3)




y3x<-c(4,5,5,7,3,7,4,6,6,7,5,8,4,7,8,8,3,6,6,7,3,6,4,7,4,6,6,7,4,4,5,7)
y3w<-c(4,5,5,6,4,7,4,7,6,8,5,7,5,7,6,8,3,6,4,7,3,5,4,6,3,6,5,6,3,7,4,7)
y3<-c(y3x,y3w)



######################################egg
```

```
##taster1 A,B
fit_eg1<-lm(y3x~A*B*C*D*E+Block)
summary(fit_eg1)
anova(fit_eg1)
est.e1<-2*fit_eg1$coefficients[-c(1,7,8,9,20,29,32)]
a.e1<-abs(est.e1)
g.e1<-length(est.e1)
#Lenth
s0.e1<-1.5*median(a.e1)
a1.e1<-a.t1[a.e1<2.5*s0.e1]
pse.e1<-1.5*median(a1.e1)
v.e1<-g.e1/3
alpha<-0.1
ga.e1<-0.5*(1-(1-alpha)^(1/g.e1))
a.e1[a.e1>qt(ga.e1,v.e1,lower.tail = F)*pse.e1]

rd.e1<-lm(y3x~A+Block)#residual plots of dong&lenth method
summary(rd.e1)
res.e1<-rd.e1$residuals
fit.e1<-rd.e1$fitted.values
plot(res.e1 ~ fit.e1,xlab="Fitted values",ylab="Residuals",main="Residuals vs
plot(y3x ~ fit.e1,xlab="Fitted values",ylab="Observed values",main="Observed
qqnorm(res.e1)
qqline(res.e1)


#Dong
m1.e1<-length(a1.e1)
s12.e1<-sum(a1.e1^2)/m1.e1
s1.e1<-sqrt(s12.e1)
a2.e1<-a.t1[a.e1<2.5*s1.e1]
m2.e1<-length(a2.e1)
s22.e1<-sum(a2.e1^2)/m2.e1
s2.e1<-sqrt(s22.e1)
a.e1>s2.e1*qt(ga.e1,m2.e1,lower.tail = F)
a.e1[a.e1>s2.e1*qt(ga.e1,m2.e1,lower.tail = F)]

rd.e1<-lm(y3x~A+B+Block)#residual plots of dong&lenth method
summary(rd.e1)
res.e1<-rd.e1$residuals
```

```
fit.e1<-rd.e1$fitted.values
plot(res.e1 ~ fit.e1,xlab="Fitted values",ylab="Residuals",main="Residuals vs
plot(y3x ~ fit.e1,xlab="Fitted values",ylab="Observed values",main="Observed
qqnorm(res.e1)
qqline(res.e1)


#trans




ta1.t1<-lm(y3x~A*B*C*D*E+Block-A:B:C:D:E)
ta1.bc1<-boxcox(ta1.t1)
ta1.bc1$x[which.max(ta1.bc1$y)]#no way

ta1.t2<-lm(y3x~(A+B+C+D+E)^3+Block)
summary(ta1.t2)
resi<-ta1.t2$residuals
fitt<-ta1.t2$fitted.values
plot(resi ~ fitt,xlab="Fitted values",ylab="Residuals",main="Residuals vs fit
plot(y3x ~ fitt,xlab="Fitted values",ylab="Observed values",main="Observed va
qqnorm(resi)
qqline(resi)
ta1.bc2<-boxcox(ta1.t2)
(l2<-ta1.bc2$x[which.max(ta1.bc2$y)])
y.ta1<-y3x^l2
model.ta1<-lm(y.ta1~(A+B+C+D+E)^3+Block)
summary(model.ta1)
anova(model.ta1)
resid.ta1 <- model.ta1$residuals#residual plots of transformation
fitted.ta1 <- model.ta1$fitted.values
fit.ta1<-fitted.ta1^(1/l2)
plot(resid.ta1 ~ fitted.ta1,xlab="Fitted values",ylab="Residuals",main="Resid
plot(y3x ~ fit.ta1,xlab="Fitted values",ylab="Observed values",main="Observed
qqnorm(resid.ta1)
qqline(resid.ta1)

coef.e1<-ta1.t2$coefficients[-c(1,7,8,9,20,29)]
g.e1<-23
v.e1<-5
```

```
M.e1<-4.1
se.e1<-0.20010
coef.e1[abs(coef.e1)>M.e1*se.e1]

ta1.t3<-lm(y3x~(A+B+C+D+E)^2+Block)
resi<-ta1.t3$residuals
fitt<-ta1.t3$fitted.values
plot(resi ~ fitt,xlab="Fitted values",ylab="Residuals",main="Residuals vs fit
plot(y3x ~ fitt,xlab="Fitted values",ylab="Observed values",main="Observed va
qqnorm(resi)
qqline(resi)
ta1.bc3<-boxcox(ta1.t3)
(l2<-ta1.bc3$x[which.max(ta1.bc3$y)])
y.ta1<-y3x^l2
model.ta1<-lm(y.ta1~(A+B+C+D+E)^2+Block)
summary(model.ta1)
anova(model.ta1)
resid.ta1 <- model.ta1$residuals#residual plots of transformation
fitted.ta1 <- model.ta1$fitted.values
fit.ta1<-fitted.ta1^(1/l2)
plot(resid.ta1 ~ fitted.ta1,xlab="Fitted values",ylab="Residuals",main="Resid
plot(y3x ~ fit.ta1,xlab="Fitted values",ylab="Observed values",main="Observed
qqnorm(resid.ta1)
qqline(resid.ta1)
coef.e2<-ta1.t3$coefficients[-c(1,7,8,9)]
g.e2<-15
v.e2<-13
M.e2<-3.1
se.e2<-0.17442
coef.e2[abs(coef.e2)>M.e2*se.e2]

ta1.t4<-lm(y3x~A+B+C+D+E+Block)
ta1.bc4<-boxcox(ta1.t4)
(l2<-ta1.bc4$x[which.max(ta1.bc4$y)])
y.ta1<-y3x^l2
model.ta1<-lm(y.ta1~A+B+C+D+E+Block)
summary(model.ta1)
anova(model.ta1)
resid.ta1 <- model.ta1$residuals#residual plots of transformation
fitted.ta1 <- model.ta1$fitted.values
fit.ta1<-fitted.ta1^(1/l2)
```

```
plot(resid.ta1 ~ fitted.ta1,xlab="Fitted values",ylab="Residuals",main="Resid
plot(y3x ~ fit.ta1,xlab="Fitted values",ylab="Observed values",main="Observed
qqnorm(resid.ta1)
qqline(resid.ta1)
```

```
##taster2 A,D,E
fit_eg2<-lm(y3w~A*B*C*D*E+Block)
summary(fit_eg2)
anova(fit_eg2)
est.e2<-2*fit_eg2$coefficients[-c(1,7,8,9,20,29,32)]
a.e2<-abs(est.e2)
g.e2<-length(est.e2)
#Lenth
s0.e2<-1.5*median(a.e2)
a1.e2<-a.e2[a.e2<2.5*s0.e2]
pse.e2<-1.5*median(a1.e2)
v.e2<-g.e2/3
alpha<-0.1
ga.e2<-0.5*(1-(1-alpha)^(1/g.e2))
a.e2[a.e2>qt(ga.e2,v.e2,lower.tail = F)*pse.e2]

rd.e2<-lm(y3w~A+Block)#residual plots of dong&lenth method
summary(rd.e2)
res.e2<-rd.e2$residuals
fit.e2<-rd.e2$fitted.values
plot(res.e2 ~ fit.e2,xlab="Fitted values",ylab="Residuals",main="Residuals vs
plot(y3w ~ fit.e2,xlab="Fitted values",ylab="Observed values",main="Observed
qqnorm(res.e2)
qqline(res.e2)


#Dong
m1.e2<-length(a1.e2)
s12.e2<-sum(a1.e2^2)/m1.e2
s1.e2<-sqrt(s12.e2)
a2.e2<-a.e2[a.e2<2.5*s1.e2]
```

```
m2.e2<-length(a2.e2)
s22.e2<-sum(a2.e2^2)/m2.e2
s2.e2<-sqrt(s22.e2)
a.e2>s2.e2*qt(ga.e2,m2.e2,lower.tail = F)
a.e2[a.e2>s2.e2*qt(ga.e2,m2.e2,lower.tail = F)]

rd.e2<-lm(y3w~A+E+Block)#residual plots of dong&lenth method
summary(rd.e2)
res.e2<-rd.e2$residuals
fit.e2<-rd.e2$fitted.values
plot(res.e2 ~ fit.e2,xlab="Fitted values",ylab="Residuals",main="Residuals vs
plot(y3w ~ fit.e2,xlab="Fitted values",ylab="Observed values",main="Observed
qqnorm(res.e2)
qqline(res.e2)


#trans



ta2.t1<-lm(y3w~A*B*C*D*E+Block-A:B:C:D:E)
ta2.bc1<-boxcox(ta2.t1)
ta2.bc1$x[which.max(ta2.bc1$y)]#no way

ta2.t2<-lm(y3w~(A+B+C+D+E)^3+Block)
resi<-ta2.t2$residuals
fitt<-ta2.t2$fitted.values
plot(resi ~ fitt,xlab="Fitted values",ylab="Residuals",main="Residuals vs fit
plot(y3x ~ fitt,xlab="Fitted values",ylab="Observed values",main="Observed va
qqnorm(resi)
qqline(resi)
ta2.bc2<-boxcox(ta2.t2)
(l2<-ta2.bc2$x[which.max(ta2.bc2$y)])
y.ta2<-y3w^l2
model.ta2<-lm(y.ta2~(A+B+C+D+E)^3+Block)
summary(model.ta2)
anova(model.ta2)
resid.ta2 <- model.ta2$residuals#residual plots of transformation
fitted.ta2 <- model.ta2$fitted.values
fit.ta2<-fitted.ta2^(1/l2)
plot(resid.ta2 ~ fitted.ta2,xlab="Fitted values",ylab="Residuals",main="Resid
```

```
plot(y3w ~ fit.ta2,xlab="Fitted values",ylab="Observed values",main="Observed
qqnorm(resid.ta2)
qqline(resid.ta2)



ta2.t3<-lm(y3w~(A+B+C+D+E)^2+Block)
resi<-ta2.t3$residuals
fitt<-ta2.t3$fitted.values
plot(resi ~ fitt,xlab="Fitted values",ylab="Residuals",main="Residuals vs fit
plot(y3x ~ fitt,xlab="Fitted values",ylab="Observed values",main="Observed va
qqnorm(resi)
qqline(resi)
ta2.bc3<-boxcox(ta2.t3)
(l2<-ta2.bc3$x[which.max(ta2.bc3$y)])
y.ta2<-y3w^l2
model.ta2<-lm(y.ta2~(A+B+C+D+E)^2+Block)
summary(model.ta2)
anova(model.ta2)
resid.ta2 <- model.ta2$residuals#residual plots of transformation
fitted.ta2 <- model.ta2$fitted.values
fit.ta2<-fitted.ta2^(1/l2)
plot(resid.ta2 ~ fitted.ta2,xlab="Fitted values",ylab="Residuals",main="Resid
plot(y3w ~ fit.ta2,xlab="Fitted values",ylab="Observed values",main="Observed
qqnorm(resid.ta2)
qqline(resid.ta2)#this

coef.e2<-model.ta2$coefficients[-c(1,7,8,9)]
g.e2<-15
v.e2<-13
M.e2<-3.1
se.e2<-0.0092236
coef.e2[abs(coef.e2)>M.e2*se.e2]


ta2.t4<-lm(y3w~A+B+C+D+E+Block)
resi<-ta2.t4$residuals
fitt<-ta2.t4$fitted.values
plot(resi ~ fitt,xlab="Fitted values",ylab="Residuals",main="Residuals vs fit
plot(y3x ~ fitt,xlab="Fitted values",ylab="Observed values",main="Observed va
qqnorm(resi)
```

```
qqline(resi)
ta2.bc4<-boxcox(ta2.t4)
(l2<-ta2.bc4$x[which.max(ta2.bc4$y)])
y.ta2<-y3w^l2
model.ta2<-lm(y.ta2~A+B+C+D+E+Block)
summary(model.ta2)
anova(model.ta2)
resid.ta2 <- model.ta2$residuals#residual plots of transformation
fitted.ta2 <- model.ta2$fitted.values
fit.ta2<-fitted.ta2^(1/l2)
plot(resid.ta2 ~ fitted.ta2,xlab="Fitted values",ylab="Residuals",main="Resid
plot(y3w ~ fit.ta2,xlab="Fitted values",ylab="Observed values",main="Observed
qqnorm(resid.ta2)
qqline(resid.ta2)
coef.t<-ta2.t4$coefficients[-c(1,7,8,9)]
N<-32
g.t<-5
v.t<-23
M<-2.46
se<-0.12208
coef.t[abs(coef.t)>M*se]


###as replication
a<-rep(A,2)
b<-rep(B,2)
c<-rep(C,2)
d<-rep(D,2)
e<-rep(E,2)
block<-rep(Block,2)
taster<-c(rep('1',32),rep('2',32))
d<-data.frame(y3,a,b,c,d,e,block,taster)

fit_eg<-lm(y3~a*b*c*d*e+block+taster,data=d)
summary(fit_eg)
anova(fit_eg)
est.effects3<-2*fit_eg$coefficients[-c(1,7,8,9,10,21,30,33)]
res3 <- fit_eg$residuals#residual plots of transformation
fit3 <- fit_eg$fitted.values
plot(res3~fit3,xlab="Fitted values",ylab="Residuals",main="Residuals vs fitte
plot(y3 ~ fit3,xlab="Fitted values",ylab="Observed values",main="Observed val
```

```
qqnorm(res3)
qqline(res3)
#smm
v3<-31
g3<-length(est.effects3)
se3.est<-0.08531*2
M30<-3.1
est.effects3[abs(est.effects3)>M30*se3.est]


qqnorm(y3)
qqnorm(est.effects3)
qqline(est.effects3)
#transformation




rep.bc1<-boxcox(fit_eg,data=d)
(l3<-rep.bc1$x[which.max(rep.bc1$y)])
y.e<-y3^l3
da<-data.frame(y.e,a,b,c,d,e,block,taster)
model.e<-lm(y.e~a*b*c*d*e+block+taster,data = da)
summary(model.e)
anova(model.e)
resid.e <- model.e$residuals#residual plots of transformation
fitted.e <- model.e$fitted.values
fit.e<-fitted.e^(1/l3)
plot(resid.e ~ fitted.e,xlab="Fitted values",ylab="Residuals",main="Residuals
plot(y3 ~ fit.e,xlab="Fitted values",ylab="Observed values",main="Observed va
qqnorm(resid.e)
qqline(resid.e)


run<-c(run1,run2+8,run3+16,run4+24)
plot(run,res3[1:32],xlab = "run orders",ylab = "residuals",main = "taster1")
plot(run,res3[33:64],xlab = "run orders",ylab = "residuals",main = "taster2")
abline(v=8.5,col="darkorange",lty=3)
abline(v=16.5,col="darkorange",lty=3)
abline(v=24.5,col="darkorange",lty=3)
```