Center for European, Governance and Economic Development Research
cege

Discussion Papers

Number 343 – March 2018

MINIMIZING LEARNING BEHAVIOR IN REPEATED REAL-EFFORT TASKS

Volker Benndorf, Holger A. Rau, Christian Sölch

GEORG-AUGUST-UNIVERSITÄT GÖTTINGEN

ISSN: 1439-2305

Minimizing Learning Behavior in Repeated Real-Effort Tasks*

Volker Benndorf[†1], Holger A. Rau[‡2], and Christian Sölch[§3]

[1]Goethe University Frankfurt
[2]University of Mannheim, University of Göttingen
[3]University of Erlangen-Nürnberg

March 2018

Abstract

In this paper, we discuss learning behavior and the heterogeneity of subjects' ability to perform in real-effort tasks. Afterwards, we present a novel variant of Erkal et al.'s (2011) encryption real-effort task which aims to minimize learning behavior in repeated settings. In the task, participants encrypt words into numbers. In our variant, we apply a double-randomization mechanism to minimize learning and heterogeneity. Existing experiments with repeated real-effort tasks find a performance increase of 12-14% between the first and second half. By contrast, our task mitigates learning behavior down to 2% between the first and second half. The data show that subjects show a small heterogeneity in performance.

JEL Classification numbers: C90, C91.

Keywords: Experimental Methods, Learning Behavior, Real Effort.

---

*We thank the participants of the European ESA Conference 2014, and seminar audiences at the University of Duesseldorf, at the University of Erlangen-Nürnberg, and at the University of Göttingen for helpful comments and criticism. Financial support by Emerging Fields Initiative of the University of Erlangen-Nuremberg and by DICE is gratefully acknowledged. We thank Brice Corgnet, Catherine Eckel, Joerg Oechssler, Emmanuel Peterlé, and Fabian Winter for helpful comments. We especially want to thank Nikos Nikiforakis for detailed comments and for sharing with us the data of Cason et al. (2011).

[†]Theodor-W.-Adorno Platz 4, 60629 Frankfurt am Main, Germany. E-Mail: *benndorf@econ.uni-frankfurt.de*

[‡]Corresponding Author, L7, 3-5, 68131 Mannheim, Germany. E-Mail: *holger.rau@uni-mannheim.de*

[§]Lange Gasse 20, 90403 Nürnberg, Germany. E-Mail: *christian.soelch@fau.de*

1 Introduction

In recent years, real-effort tasks enjoyed increasing popularity in experimental economics (Lezzi et al., 2015). These procedures allow the experimenter to pay subjects based on their performance, which is more realistic than allocating "windfall" money to them. However, applying real work tasks may cause uncontrolled variation among subjects.

Examples are settings where the work tasks are repeated to control for subjects' performance. This may lead to a performance increase in later periods, as subjects may be able to use their experience from prior periods.[1] Think of a repeated real-effort experiment with two parts of ten periods. Assume that correctly solved puzzles are rewarded by a low piece rate in the first ten periods and by a high piece rate in the last ten periods. If subjects learn doing the task, they may increase performance in the first ten periods, where they constantly receive the low piece rate. If the high piece rate leads to an incentive effect, it is possible that the learning behavior overlays the treatment effect. Hence, it may not only be complicated to isolate the two effects, but also it is possible that the learning effect may blow out the treatment effect. A similar observation is made by Araujo et al. (2016), who test incentive effects in a repeated real-effort setting with the slider task of Gill and Prowse (2011). The paper focuses on a between-subjects experiment with treatments of different piece rates. Independently of the piece rate, subjects always show a significant learning effect and a similar performance across treatments. Another problem of uncontrolled variation maybe heterogeneity in subjects' ability to perform the real-effort task.[2] This may be emphasized through learning behavior when the task is repeated. Examples are settings which focus on subjects' work motivation (Benndorf et al., 2017), or on tullock-like tournaments (Dechenaux et la., 2015).[3]

In this paper, we present evidence concerning learning behavior and heterogeneity in performance, in three widely used tasks. To amend these problems, we report data of a new variant of one of these tasks, showing that variance and learning are minimized. The task builds on Erkal et al.'s (2011) word-encryption setting, where subjects encode combinations of letters to numbers. The innovation of our variant is a double-randomization mechanism, which is applied whenever a puzzle is correctly solved. The task not only shuffles the letter- and number allocation in the table, but also the position of the letters.

[1] In this paper, we focus on minimizing learning behavior in the task, i.e., the fact that subjects perform better in later periods. By contrast, we do not analyze learning in a strategic game, which occurs when subjects get a better understanding of the payoff functions and of the strategic interaction (e.g., Nagel, 1995; Huck et al., 1999).

[2] We thank Nikos Nikiforakis for raising this point.

[3] Heterogeneity in subjects' ability may even be desirable when real-effort tasks are used to endogenize money, e.g., to establish property rights (e.g., Cherry et al., 2002; Erkal et al., 2011; Heinz et al., 2012).

We report data of a repeated experiment and show that the task minimizes learning and performance heterogeneity. Between the first half and second half of our data, we find that subjects only increase performance by 2%.

2 Learning in Real-Effort Tasks and Power Analysis

We review learning behavior in three popular real-effort tasks: the *counting-numbers task*, the *slider task*, and the *word-encryption task*. Finally, we introduce our modification of the word-encryption task. We always report one-sided $p-values$ and conduct an ex-post power analysis,[4] where we report the effect size using Cohen's d. We formulate a target power of $\Pi = 1 - \beta$, where β is the probability of a type II error.

2.1 Counting-Numbers Task

Abeler et al. (2011) introduce a z-Tree based task where subjects receive a grid full of numbers and have to count the number of zeros. After subjects have entered an answer, they receive a new puzzle. The task counts the number of correctly solved puzzles. Table 1 displays an example of a possible grid.

11011001010
01011101100
11101000111
10100111010
00101010110

Table 1: Schematic representation of a counting-numbers puzzle

The counting-numbers task is simple to understand and to implement. It does not require preexisting knowledge. It is tedious and may adequately resemble work effort. Lezzi et al. (2015) analyze the task in a repeated setting with 10 periods. Subjects are given two minutes in each period and have to count the number of ones in a 5 by 5 grid. Table 2 overviews their findings conditional on the half of the experiment (periods 1–5 vs. periods 6–10) and subjects' gender.

The paper finds a highly significant increase of 14% in the number of correctly solved grids between periods 1–5 (10.43) and periods 6–10 (11.91) (Wilcoxon matched-pairs test, $p = 0.006$, $d = 0.24$, $\Pi = 0.99$). Similar evidence is found by Vranceanu et al. (2015). In Lezzi et al. (2015), learning occurs for men and women. However, the authors find no significant gender differences in performance, which is in line with the one-shot data

[4]We use the g*Power software tool (Faul et al. 2007).

	obs.	mean	std. dev.	min	max
periods 1–5					
male	125	10.93	5.97	0	23
female	135	9.96	6.13	0	25
all data	260	10.43	6.06	0	25
periods 6–10					
male	125	12.17	6.15	0	24
female	135	11.68	5.93	0	25
all data	260	11.91	6.03	0	25

Table 2: Mean number of correctly solved grids in the data of Lezzi et al. (2015)

of Grosch and Rau (2017). Focusing on heterogeneity in performance, we find that the standard deviation is high in the counting-numbers task, but does not change between the first (6.06) and the second half (6.03) of the experiment. The task seems to be appropriate for settings where the experimenter intends to endogenize money in a one-shot scenario.

2.2 The Slider Task

Gill and Prowse (2011; 2012) propose a z-Tree based real-effort task, where subjects have to adjust sliders to the middle of a bar. The task is simple to communicate and to understand. There is no scope for guessing. Subjects are presented a screen of 48 sliders. The sliders are arranged such that no slider is exactly placed under another one. Sliders can be adjusted by mouse clicks. A counter next to the slider indicates the current position, which is represented by a number. It changes its value (from 0 to 100), whenever the slider is adjusted. A puzzle is correctly solved when the slider is exactly adjusted to the middle of the bar. One period typically lasts 120 seconds. Figure 1 displays a slider's initial position, as well as the position where it has to be adjusted to.

Figure 1: Schematic representation of a slider

Table 3 presents the data of Gill and Prowse (2011). It not condition on gender, as the data of Gill and Prowse (2011) does not contain this information. We find a significant performance increase of 8% between the first and second half of the experiment (Wilcoxon matched-pairs test, $p < 0.001$, $d = 0.28$, $\Pi = 0.68$).

	obs.	mean	std. dev.	min	max
periods 1–5	60	23.67	6.14	0	38
periods 6–10	60	25.53	7.01	0	40
all data	60	24.60	6.65	0	40

Table 3: Mean number of correctly solved sliders in the data of Gill and Prowse (2011)

The learning effect is also found by Lezzi et al. (2015), who find in a repeated slider setting with ten periods that learning is even more pronounced. That is, subjects significantly increase their performance by 14% between the first and second half. The performance increase over time has important implications. Araujo et al. (2016) point out that learning may complicate incentive effects in the slider task. The paper focuses on a repeated between-subjects experiment with ten periods and different piece rates ($0.005, $0.02, $0.08) across treatments. Regressions show that subjects show for all piece rates a significant learning effect and average output does not change when incentives are altered. They solve in period 1 approximately 24 sliders[5] and increase their performance to approximately 28[6] in period 10. The authors report that despite a 16-fold increase in the piece rate, the performance increases only by one slider. The observed incentive effect corresponds to less than a quarter of the average learning effect. Araujo et al. (2016) emphasize that desirable tasks should be able to demonstrate incentive effects, which are large enough to survive uncontrolled variation (e.g., learning behavior) within the task.

Turning to heterogeneity in performance, we find that the standard deviation of the slider task is large and increases between periods 1–5 (6.14) and periods 6–10 (7.01). This indicates that the learning bias may also emphasize performance heterogeneity.

2.3 The Word Encryption Task

Erkal et al. (2011) and Cason et al. (2011) present a z-Tree word encryption task where subjects have to encode words to numbers. The task is very simple and easy to communicate to them. The numbers in the task are given by an allocation table which allocates a one-digit or two-digit number to each letter of the alphabet. The letters are sorted in the order of the alphabet. A puzzle is solved when the correct number of all letters was entered.

Cason et al. (2011) apply the task to test whether real-effort investments can inhibit equilibrium convergence of experimental markets. In one treatment (costs treatment) pro-

[5]Performance is 24.2 in the $0.005 and $0.02 treatments. Whereas, it is 24.2 in the $0.08 treatment.
[6]In period 10 of the $0.005 treatment it is 28.6, whereas it is 28.9 in the $0.02 and $0.08 treatments.

duction costs of sellers are allocated based on their relative performance in the task and a random productivity shock, which determines sellers' final points and cost of production.[7]

	costs treatment					values treatment				
	obs.	mean	std. dev.	min	max	obs.	mean	std. dev.	min	max
part 1										
male	18	32.50	8.32	15	49	14	33.79	4.93	23	40
female	7	36.00	3.87	31	40	11	31.27	5.26	25	43
all data	25	33.48	7.44	15	49	25	32.68	5.13	23	43
part 2										
male	18	39.22	9.43	22	59	14	37.57	6.64	21	49
female	7	46.71	6.47	38	53	11	38.55	5.82	31	50
all data	25	41.32	9.23	22	59	25	38.00	6.18	21	50
part 3										
male	18	43.78	12.09	22	67	14	38.57	5.73	27	51
female	7	54.57	7.59	43	63	11	40.19	8.48	21	51
all data	25	46.80	11.93	22	67	25	39.28	6.96	21	51

Table 4: Mean number of correctly encoded words in the data of Cason et al. (2011)

In the values treatment sellers' effort can increase buyers' surplus. If a seller meets the target of 35 correctly solved words, then the buyers' values increase.[8] The experiment lasts 30 periods and the task is repeated every 10 periods. Each time when it starts, subjects are given seven minutes to encrypt words. The allocation table always uses the same allocation of numbers. Table 4 reports subjects' performance conditioned on the two treatments.

Subjects significantly increase performance between part 1 and part 3 by 40% in the costs treatment and by 20% in the values treatment (both treatments: Wilcoxon matched-pairs tests, $p < 0.001$, $d = 1.28$ (costs t.), $d = 1.02$ (values t.), $\Pi > 0.99$).[9] Learning occurs for men, who increase their performance by 35% in the costs treatment (Wilcoxon matched-pairs test, $p < 0.001$, $d = 1.05$, $\Pi > 0.99$) and by 14% in the values

[7]The seller with the highest number of points is assigned the lowest production cost, the seller with the second highest number of points is assigned the second lowest production cost, etc.

[8]If three or more sellers solve 35 words then the demand schedule is the same as in the costs treatment.

[9]Significant performance differences can be found between part 1 and part 2 (both treatments: Wilcoxon matched-pairs tests, $p < 0.001$, $d = 0.92$ (costs t.), $d = 0.93$ (values t.), $\Pi > 0.99$) and between part 2 and part 3 for the costs treatment (Wilcoxon matched-pairs test, $p < 0.001$, $d = 0.96$, $\Pi = 0.77$), but not for the values treatment (Wilcoxon matched-pairs test, $p = 0.234$, $d = 0.19$, $\Pi = 0.23$).

treatment (Wilcoxon matched-pairs test, $p = 0.016$, $d = 0.89$, $\Pi = 0.97$). Learning is even more pronounced for women who increase effort by 52% in the costs treatment (Wilcoxon matched-pairs test, $p = 0.018$, $d = 2.82$, $\Pi > 0.99$) and by 29% in the values treatment (Wilcoxon matched-pairs test, $p = 0.005$, $d = 1.20$, $\Pi = 0.86$). The learning effects may be due to the fact that subjects memorize the letter and number allocations in the real effort task. It may also play a role that letters are positioned in the real order of the alphabet and that the positions never change. Another factor may be related to the external rewards for high performance.

The standard deviation is in the medium range. It varies between 5.13 and 11.93. The task may be a good choice for settings which are not interested in the task performance as an outcome variable. For instance, settings which aim to endogenize outcomes, e.g., endoments (Erkal et al., 2011), or production costs (Cason et al., 2011).

3 Experimental Design of the WEDR Task

We introduce a new real-effort task which extends Erkal et al. (2011).[10] The z-Tree code (in English language) can be downloaded at:
http://www.uni-goettingen.de/de/document/download/311cacb804d57bbb12d2e7534d51a9ef.ztt/wedr.ztt[11]

The task encompasses the advantages of Erkal et al.'s (2011) task. There is no scope for guessing the results. Subjects in our variant have to encrypt combinations of three letters into three-digit numbers (see Table below). Participants are presented two rows: one which displays a word to encrypt ("word") and another one, where the solution has to be entered ("code"). Below, subjects are shown an encryption table which allocates numbers to letters. The grid always displays all 26 capital letters of the German alphabet except mutations.[12] Subjects have to type in the correct three-digit numbers of each letter in the "code" row below the letter.

word:	**Z**	**N**	**T**

code:	113	154	

[10]Other papers applying similar encryption tasks are: Cason et al. (2011), Nikiforakis et al. (2012), Charness et al. (2013), McDonald et al. (2013).
[11]We friendly ask all people using the WEDR task to cite this paper.
[12]For reasons of space only 15 allocations are presented in the example of Table 8.

encryption table:

B	T	R	S	U	Z	F	N	C	Y	V	X	H	Y	K
384	118	201	543	386	113	980	154	745	265	432	262	110	960	245

Table 5: Example of a problem in the real-effort task.

After all three letters are encoded, subjects press a submit button. They are informed about the total number of correctly solved puzzles in the current period. The allocation table randomly allocates a new number to all letters, whenever subjects have correctly encrypted a word. At the same time, the positions of all letters are randomly re-arranged. This double randomization is a special feature of our task. The idea is that this additionally complicates learning behavior within the task. We thus call our task: *Word Encryption task with Double Randomization* (henceforth: *WEDR* task).

When subjects enter a wrong answer they are informed by the computer program. Then, the number allocations and the locations of the letters will not be shuffled, until subjects make a correct input. After the end of two minutes the task automatically stops and inputs are not possible anymore. After the end of each period the computer program automatically proceeds to the next round. A further change in the *WEDR* task is that our "words" only consist of three letters. We let the computer program chose fictitious combinations of three letter combinations. We believe that this complicates working in the task, which should additionally mitigate learning behavior.

Procedures

We conducted seven sessions between December 2013 and January 2015 in the LERN laboratory at the University of Erlangen-Nürnberg. One session typically encompassed 32 subjects.[13] In total, we have 219 independent observations (108 men, 111 women). Before the experiment started, subjects received a set of written instructions explaining the usage of the real-effort task. After all subjects confirmed that they understood the functioning of the task, we started a trial period.[14] The participants were asked to solve exactly ten puzzles of the task without being paid. After all subjects finished that, we provided them with a new set of instructions. They were told that they participate in ten periods which each last 120 seconds. They were informed hat they would receive a piece rate of €0.08 for each correctly solved puzzle. In the instructions it was explained

[13]In one case only 28 subjects showed up. In another case a computer of a participant crashed during the experiment. We therefore dropped this observation.

[14]The purpose of the trial period is to make subjects familiar with the task. This procedures may also help to mitigate learning behavior.

that the experiment will consist of two parts and that the current instructions only cover the first part. They knew that they will receive the new instructions after the end of period 10. The second part belongs to a second experiment, where we study the impact of remuneration changes (Benndorf et al., 2017). On average the first part lasted 35 minutes. The experiment was programmed in z-Tree (Fischbacher, 2007). The subject pool mainly consisted of economic students which were recruited with ORSEE (Greiner, 2015). In the first part subjects earned on average €7.86.

4 Results

We first discuss the development over time of subjects' performance. In a next step, we apply non-parametric test methods to analyze performance changes between the first half (periods 1–5) and second half (periods 6–10) of the experiment. Finally, we conduct regression analyses controlling for the impact of demographics on subjects' performance in the task.

4.1 Word Encryption with Double Randomization (WEDR)

Figure 2 presents subjects' performance in periods 1-10 of the *WEDR* task. The solid line represents subjects' mean performance over time. The diagram also conditions on the mean performance of males (black dashed line) and females (grey dashed line).

On average participants solve 9.82 words correctly.[15] The development of performance is flat and it only moderately increases over time. It can be seen that the real-effort task apparently mitigates learning behavior of subjects. All three time lines do not show a conspicuous increase over time. It turns out that women (10.09) outperform men (9.55) (Mann-Whitney test, $p = 0.010$, $d = 0.32$, $\Pi = 0.74$).[16] Table 6 displays the effort development of males and females between periods 1–5 and periods 6–10. Focusing on effort between the first and second half, we find a weak performance increase of 2%. Although, learning behavior is clearly mitigated, the increase turns out to be significant (Wilcoxon matched-pairs test, $p < 0.001$, $d = 0.14$, $\Pi = 0.67$). However, the very small Cohen's d emphasizes that the effect size is weak. A closer look reveals that both genders increase their performance by 2% (Wilcoxon matched-pairs tests, both genders: $p < 0.001$, (males: $d = 0.15$, $\Pi = 0.43$; females: $d = 0.14$, $\Pi = 0.45$)).

We find that the standard deviation in the *WEDR* task is pretty small. Importantly, it does not increase over time (periods 1–5: 1.49; periods 6–10: 1.69). We summarize that

[15] See Table 8 in the appendix for a detailed overview of subjects' performance over time.
[16] This confirms the findings of Majeres (1983).

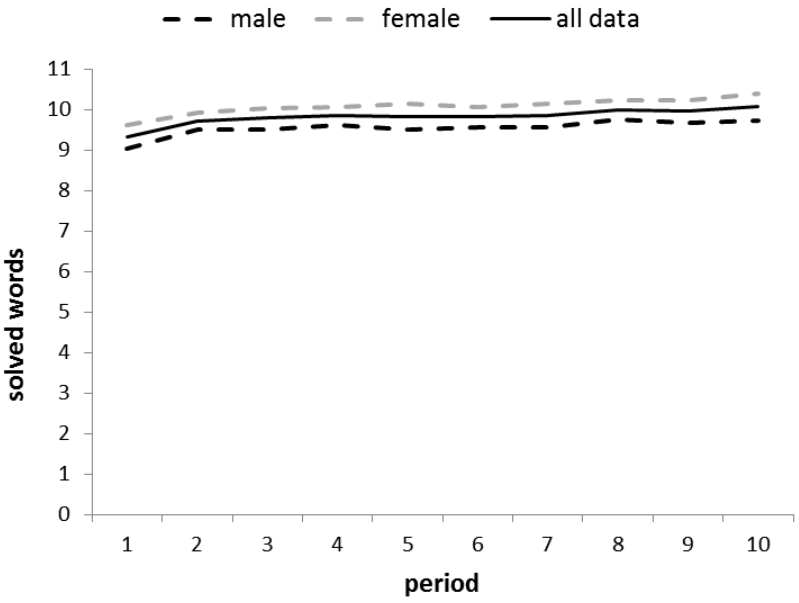


Figure 2: mean of correctly solved words

	obs.	mean	std. dev.	min	max
periods 1–5					
male	108	9.44	1.50	0	14
female	111	9.96	1.66	6	14
all data	219	9.71	1.49	0	14
periods 6–10					
male	108	9.67	1.63	4	14
female	111	10.21	1.71	4	15
all data	219	9.94	1.69	4	15

Table 6: Mean number of correctly solved grids in the *WEDR* task

the double randomization mechanism clearly mitigates subjects' learning behavior over time. Although, the participants show a significantly higher performance in the second half, we find that the increase is small and not economically significant. The small and stable standard deviation shows that the task successfully prevents uncontrolled variation among subjects. Hence, the task may be applied in repeated (within subjects) settings aiming at the analysis of subjects' work motivation. To get deeper insights on subjects' performance we focus on panel regression analyses in the next section.

4.2 Regression Analysis

In this section we control with regression analyses for the impact of different demographics on subjects' performance. We elicited the data in a post-experimental questionnaire. Table 7 presents the results of random-effects panel regressions.

	mean performance	
	(1)	(2)
period	0.128***	0.128***
	(0.031)	(0.031)
period squared	-0.006**	-0.003**
	(0.003)	(0.003)
female		0.423**
		(0.195)
econ		0.478*
		(0.262)
age		0.013
		(0.026)
fun		0.124***
		(0.038)
constant	9.368***	7.734***
	(0.122)	(0.698)
Wald chi^2	73.17	96.16
obs.	2190	2190
Standard errors in parentheses		
*** p<0.01, ** p<0.05, * p<0.1		

Table 7: OLS regression on average performance.

Model (1) controls for time dynamics, i.e., *period* and *period squared* analyze the effects of learning in the course of the experiment. In model (2) we incorporate control variables for demographics. We integrate two dummies: *female* which controls for subjects' gender (0 = male; 1 = female) and *econ* which controls whether subjects are economic students (0 = non-econ student; 1 = econ student). The model controls for subjects' *age* in years. We also include subjects' self-reported perception of fun while doing the task (*fun*). The variable was measured in a questionnaire, where subjects had to state on 10-point likert scale (1 = no fun; 10 = great fun) how much fun they had when doing the task.

Models (1) and (2) show that *period* is statistically significant and positive. The coefficients are small and not economically significant. It can be seen that *period squared* is negative and significant. This suggests that subjects' development of performance is not linear. The fact that *period* is positive and *period squared* is negative, emphasizes that subjects in the beginning increase performance, while learning stops after a while.

Focusing on model (2), we find that *female* is significant with a positive sign, i.e., women perform better than men. Regression (2) shows that *age* does not impact on performance. Whereas, subjects who report a higher level of fun perform better. At the same time, econ students moderately perform better.

5 Conclusion

We presented data of a real-effort task to mitigate learning and performance heterogeneity in repeated settings. Therefore, we modified Erkal et al.'s (2011) word encryption task and applied a double-randomization mechanism. It varies the coding of the letters to numbers and the positions of the letters in the allocation table.

Although, we find that subjects in the task show a moderate performance increase over time, we conclude that the task minimizes learning behavior. Our data document that subjects show a slight increase of 2% between the first half and second half. If we compare this to other real-effort tasks, we are confident that this new task provides a helpful contribution in addressing the problem of uncontrolled variation within real-effort tasks. The standard deviation in the task performance is small and stable over time. Thus, the task seems to be a good candidate to be applied in repeated settings focusing on incentive effects (Benndorf et al., 2017; Köhler et al., 2015). The data of the *WEDR* task is promising, as it show that learning biases in repeated real-effort tasks can be mitigated by applying simple modifications. The findings may stimulate the designs of repeated real-effort experiments. Applying tasks which do not suffer learning biases, may help to cleanly identify incentive effects induced by treatment changes. It is valuable to think about additional improvements in real-effort tasks.

References

[1] Abeler, J., Falk, A., Goette, L., Huffman, D. (2011), "Reference points and effort provision." American Economic Review 101, 470-492.

[2] Araujo, F. A., Carbone, E., Conell-Price, L., Dunietz, M. W., Jaroszewicz, A., Landsman, R., ... and Wilson, A. J. (2016). "The slider task: an example of restricted inference on incentive effects." Journal of the Economic Science Association, 2(1), 1-12.

[3] Benndorf, V., Rau, H.A., Sölch, C., (2017). "Gender Differences in Motivational Crowding Out of Work Performance." CeGe Discussion Paper 304.

[4] Cason, T., Gangadharan, L., Nikiforakis, N., (2011). "Can Real-Effort Investments Inhibit the Convergence of Experimental Markets." International Journal of Industrial Organization 29 (1), 97-103.

[5] Charness, G., Masclet, D., Villeval, M. C. (2013). "The dark side of competition for status. Management Science, 60(1), 38-55."

[6] Cherry, T., L., Frykblom, P., Shogren, J., F., (2002). "Hardnose the Dictator." American Economic Review 92, 1218-1221.

[7] Dechenaux, E., Kovenock, D., Sheremeta, R. M. (2015). "A survey of experimental research on contests, all-pay auctions and tournaments." Experimental Economics, 18(4), 609-669.

[8] Erkal, N., Gangadharan, L., Nikiforakis, N. (2011). "Relative Earnings and Giving in a Real-Effort Experiment." American Economic Review, 101, 3330 - 48.

[9] Faul, Franz, Edgar Erdfelder, Albert-Georg Lang, and Axel Buchner. 2007. "G^* Power 3: A flexible statistical power analysis program for the social, behavioral, and biomedical sciences," Behavior Research Methods 39 (2): 175191.

[10] Fischbacher, U. (2007). "z-Tree: Zurich Toolbox for Readymade Economic Experiments - Experimenter's Manual." Experimental Economics 10, 171-178.

[11] Gill, D., Prowse, V. (2011). "A novel Computerized Real Effort Task Based on Sliders." IZA Discussion Paper 5801.

[12] Gill, D., Prowse, V. (2012). "A structural analysis of disappointment aversion in a real effort competition." American Economic Review, 102(1), 469-503.

[13] Grosch, K., Rau, H.A. (2017) "Do Discriminatory Pay Regimes Unleash Antisocial Behavior?" CeGe Discussion Paper 315.

[14] Greiner, B. (2015). "Subject pool recruitment procedures: organizing experiments with ORSEE." Journal of the Economic Science Association, 1(1), 114-125.

[15] Heinz M., Juranek, S., Rau, H.A., (2012). "Do Women Behave More Reciprocally than Men? Gender Differences in Real Effort Dictator Games." Journal of Economic Behavior & Organization 83, 105-110.

[16] Huck, S., Normann, H.T., Oechssler, J., (1999). "Learning in Cournot oligopoly—An experiment." The Economic Journal 109, 80-95.

[17] Köhler, K., Pagel, B., Rau, H. A. (2015). "How worker participation affects reciprocity under minimum remuneration policies: Experimental evidence." CeGe Discussion paper 267.

[18] Lezzi, E., Fleming, P., Zizzo, D. J. (2015). "Does it matter which effort task you use? a comparison of four effort tasks when agents compete for a prize." Working Paper.

[19] Majeres, R. L., 1983. "Sex differences in symbol-digit substitution and speeded matching." Intelligence.7 (4), 313-327.

[20] McDonald, I., Nikiforakis, N., Olekalns, N., Sibly, H., (2013). "Social Comparisons and Reference Group Formation: Some Experimental Evidence." Games and Economic Behavior 79, 75-89.

[21] Nagel, R., (1995). "Unraveling in guessing games: An experimental study." American Economic Review 85, 1313-1326.

[22] Nikiforakis N., Noussair, C., and Wilkening, T., (2012). "Normative Conflict and Feuds: The Limits of Self-Enforcement." Journal of Public Economics 96 (9-10), 797-807.

[23] Vranceanu, R., El Ouardighi, F., Dubart, D. (2015). "Team Production with Punishment Option: Insights from a RealEffort Experiment." Managerial and Decision Economics, 36(6), 408-420.

Appendix

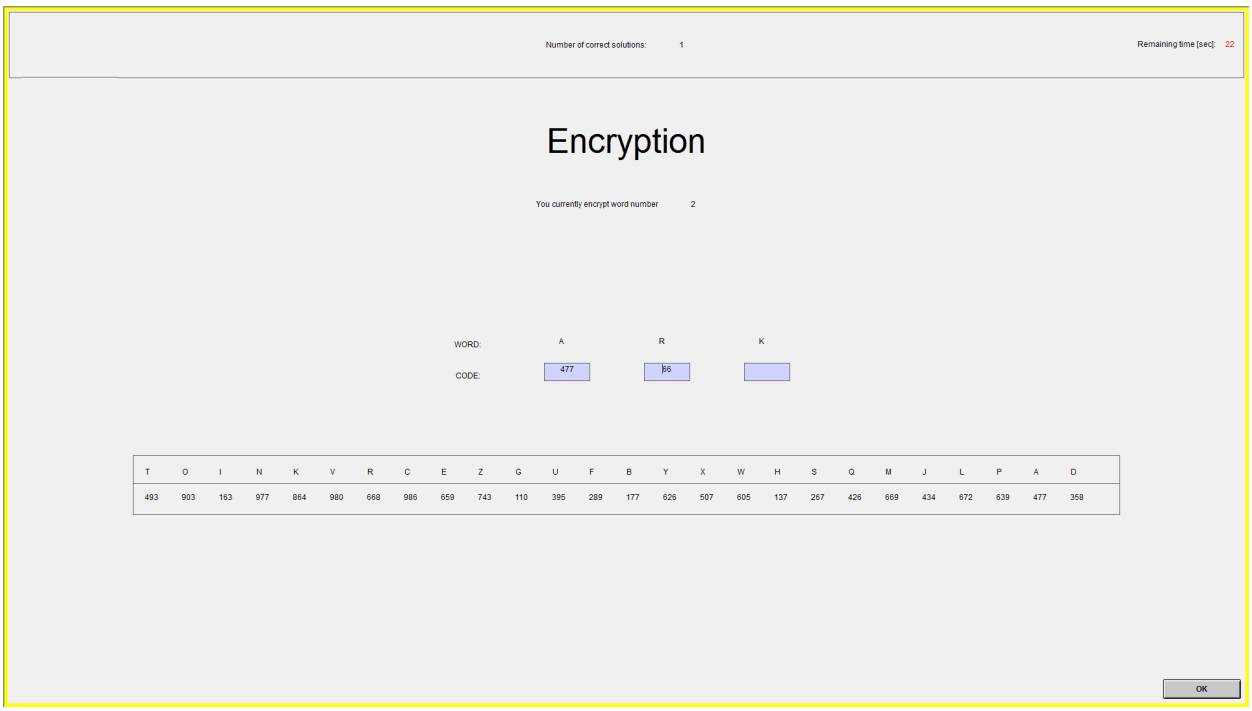


Figure 3: Screen shot of the WEDR task

	period										
	1	2	3	4	5	6	7	8	9	10	overall
male	9.04	9.50	9.52	9.62	9.52	9.57	9.57	9.75	9.69	9.74	9.55
sd	1.78	1.76	1.76	1.53	1.89	1.78	1.69	1.55	1.63	1.49	1.69
female	9.62	9.94	10.05	10.07	10.14	10.06	10.14	10.23	10.24	10.40	10.09
sd	1.73	1.61	1.68	1.57	1.68	1.67	1.79	1.72	1.64	1.76	1.69
all data	9.33	9.72	9.79	9.85	9.83	9.82	9.86	10.00	9.97	10.07	9.82
sd	1.77	1.69	1.74	1.56	1.81	1.74	1.76	1.65	1.66	1.66	1.71

Table 8: mean of correctly solved words in the 10 periods of the *WEDR* task. The table also conditions on gender (male, n = 108: female, n = 111) Standard deviations in parentheses.

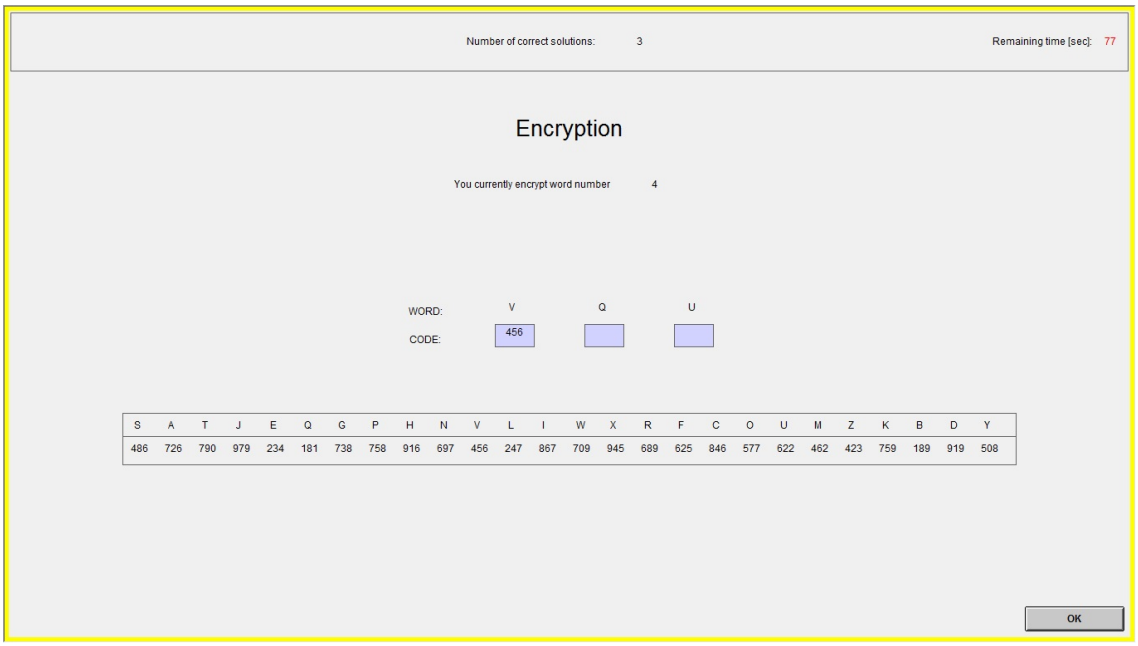
6 **Instructions** (translated from German; not intended for publication)

6.1 Instructions describing the task and the trial period

In the following experiment you have the opportunity to earn money depending on your behavior. Please turn off your mobile phone and do not talk to other participants in the experiment. It is very important that you follow these rules. If you have any question while reading these instructions or during the experiment itself, we ask you to raise your hand. We will immediately come to your desk and answer your question individually.

1. General structure of the experiment

During the experiment you have the opportunity to do a task. The task consists of encoding combinations of letters (words) into numbers. In the task, three capital letters always yield a word. You have to allocate a number to each capital letter. The encryption code can be found in a table below the corresponding letter. For that purpose, please consider the following screenshot:



In this example the participant has already encrypted three words correctly (see centered field: above). Here, the three capital letters: V, Q and U have to be encoded. The solution follows immediately from the table:

- For “V” applies: 456 (see the current entry of the participant)
- For “Q” applies: 181
- For “U” applies: 622

To make an input please click on the blue box below the first capital letter.

Furthermore, the screen (see screenshot) provides the following information:

- “Number of correct solutions” = number of correctly encrypted words.
- “Remaining time [sec]” = remaining time in the current period.
- “You currently encrypt word number” = current word to encrypt.

If all 3 numbers have been entered, please click the “OK”. The computer then checks whether all capital letters haven been encoded correctly. Only then the word is counted as correctly solved. Thereafter a new word (again consisting of three capital letters) is randomly drawn.

Furthermore, a new encryption table is randomly generated in two steps:

- The computer program randomly selects in the table a new set of three-digit numbers to be used for the encoding of the capital letters.
- Additionally, the computer program shuffles the position of the capital letters in the table. Please note that the program always uses all 26 capital letters of the German alphabet. Please note that if a new word appears, you have to click with your mouse on the first of the three blue boxes. Otherwise no input is possible!

The computer will mark (in red font) wrong inputs after pressing the OK button.

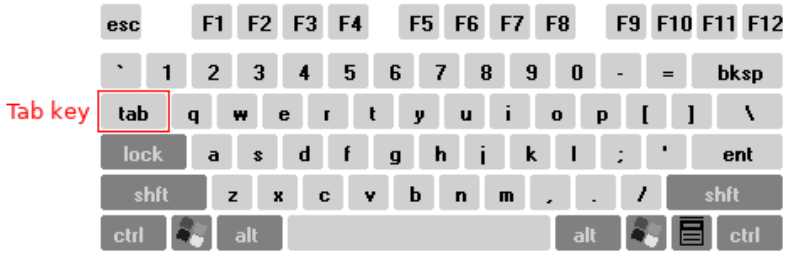
Bear in mind:

- After wrong inputs the current word to encode will not change until a correct input was made.
- However, your previous inputs (in the 3 boxes below the capital letters) will all be deleted.
- Furthermore, the table stays unaltered, meaning that the allocated numbers remain identical. Also the position of the capital letters in the table does not change.

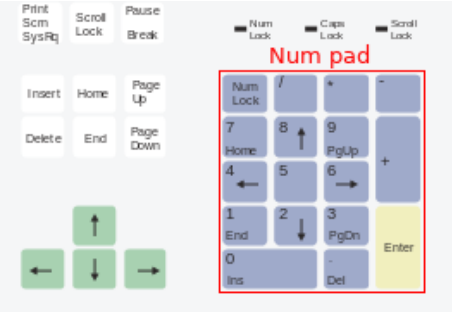
Important hints:

Please note that after having entered the three-digit number you can easily switch to the next blue box by using the tabulator key on your keyboard.

In the following picture you can see the position of the tabulator key on your keyboard:



The input of the numbers can be performed faster by using the numpad (on the right) of your keyboard. In the following picture you can see the position of the numpad on your keyboard:



2. Trial period:

The experiment starts with a trial period in which each participant has to encrypt exactly 10 words. Please note: Correct solutions do not lead to payments within the trial period. The general idea of the trial period is to make you as familiar as possible with the task before the actual experiment begins. Therefore you should take the trial period serious and try to solve the ten words as fast as possible!

Please raise your hand if you still have further questions. We will come to your desk and answer them individually.

6.2 Instructions: part one

Please note: The experiment consists of 2 parts, whereas this part encompasses 10 periods. In each of these periods you have the possibility to do the same task: encoding words to numbers.

- Each period will last exactly 2 minutes.
- After the time has expired the corresponding period will be finished. Then the program only counts the words you correctly solved to the result of the period.

In this part of the experiment, your payoff only depends on the number of correctly-solved words. You will receive 8 Cent for each correctly solved word.

It applies that:

- After each period you will see an information screen presenting the number of correctly solved words of the past period.
- The next period will automatically start 10 seconds later.

This part of the experiment will end after the end of the 10 periods. Then you will be informed on the payoff you earned in this part. Please remain seated after the end of this part. We will inform you about the further process within a short time.

Please raise your hand if you should have any further questions. We will then come to your desk and answer it individually.