Network Latency and Cloud Games: A Study Using GamingAnywhere

James Anouna, Zachary Estep, Michael French

May 4, 2014

An Interactive Qualifying Project Report:

submitted to the
Faculty of the
WORCESTER POLYTECHNIC INSTITUTE
in partial fulfillment of the requirements for the
Degree of Bachelor of Science
By

James Anouna

Zachary Estep

Michael French

Date: May 2014

Approved:

Professor Mark Claypool, Advisor

Professor David Finkel, Co-Advisor

Abstract

Thin client gaming services (services that utilize a cloud gaming model) allow consumers with low-end computers to play modern high-end video games. To do this, thin client gaming services handle the bulk of the game in the cloud, receive input from the user, and stream the visuals of the game to the user's computer. The performance on thin client games is largely based on the quality of the user's Internet connection, in which a common bottleneck is the network latency. We performed a study to analyze the relationship between network latency, performance, and quality of experience (QoE) of thin client games. We created a thin client gaming setup and introduced artificial network latency between the server and client. Participants of our study performed worse and had a lower QoE as network latency increased, with a sharp decline in performance and QoE at an intermediate latency.

Table of Contents

Chapter 1: Introduction

Thin-client gaming services enable low-end computers to run high-end video games. To achieve this, a thin-client gaming server handles the bulk of the processing, sends a video stream of the game to the client, and receives user input from the client. In theory, the experience is no different to users than playing a normal video game. The requirements to play a thin-client game are simple: have a computer at least powerful enough to display the video, and have a fast enough Internet connection to reliably stream the game. Most modern computers are able to achieve the former, while the latter is a common bottleneck in thin-client services. If a computer's connection has too much latency or packet loss or too little bandwidth, thin-client games become unplayable. The goal of this study is to determine the correlation between network latency and both quality of experience and player performance while using cloud gaming systems.

Related works show what aspects of the game are affected by a slow connection. The higher the network latency, the higher the user input delay and, depending on the thin-client service being used, the lower the frame rate. These symptoms have a negative effect on the user quality of experience, especially in the case of games requiring constant input from the user.

Network latency is the delay between two computers on a network.[1] The higher the latency between two computers, the slower the network connection. In this experiment, we investigate the effects latency has on thin-client video games. To do this, we set up a test environment between client and server computers using an open-source cloud gaming program called GamingAnywhere.[2] We studied 34 participants, each of whom played the same level of a

[1] Usually measured in round trip time, the amount of time it takes for a packet to travel from computer A to computer B then back to computer A.
[2] http://gaminganywhere.org/

game called Neverball at various artificial latencies created by a program called Dummynet. We recorded participants' performance, as well as answers to five quality of experience questions.

Players' mean performance was better at lower latencies with a sharp decrease in performance between 66ms and 150ms. Players' performance remained low on average for higher latencies. Players noted a change in their quality of experience and subjective performance that correlated with latency. As latency increased, both players' quality of experience and subjective performance decreased.

In Chapter 2 we talk about related works that influenced our study. In Chapter 3 we explain the methodology of our study. Chapter 4 is an in-depth analysis of the data obtained from our study. We summarize our findings in Chapter 5 and suggest future work in Chapter 6.

Chapter 2: Related Work

2.1: Introduction

We reviewed previous work to find which variables in thin-client games are affected by latency, and how they affect the users’ quality of experience (QoE) before conducting our study. We found two studies, *Cloud Gaming: Architecture and Performance* and *On the Quality of Service of Cloud Gaming Systems*, which provided information on the technical effects of network latency on thin-client games (Shea, 2013; Chen, 2013). We also found three studies: *Gaming in the clouds: QoE and the users' perspective*, *Are All Games Equally Cloud-Gaming-Friendly? An Electromyographic Approach*, and *Assessing measurements of QoS for global cloud computing services* that provided insight on the effects latency has on user QoE (Jarshel, 2013; Lee, 2012; Pedersen, 2011).

2.2: Latency and Thin-Client Games

We reviewed two papers to predict the effects latency has on thin-client games. Shea et al. conducted a systematic analysis of cloud gaming platforms and measured their performance (Shea, 2013). As part of performance measurements, Shea et al. introduced latency to an existing cloud gaming service called Onlive. It was found that interaction delay rose linearly with latency. Another study by Chen et al. discussed the effects of network latency, packet loss, and bandwidth on two cloud gaming services: OnLive and StreamMyGame (Chen, 2013). Frame rate was found to be affected by all three cases in OnLive. StreamMyGame’s frame rate was only affected by packet loss and bandwidth. Graphic quality was virtually unaffected in all three cases. In summary, according to these past works, thin-client services’ user input delay and frame rate are potentially affected by network latency, while graphical quality (on a frame-by-frame basis) remains the same.

2.3: Latency in Thin-Client Games and Quality of Experience

We reviewed three papers to predict the effects latency will have on users' quality of experience when playing thin-client games.

Jarschel et al. examined the effects of packet loss and latency on user quality of experience (QoE) in a controlled thin-client setup (Jarschel, 2013). Participants were asked to play videogames of slow, medium, and fast gameplay through their setup under different latency and packet-loss conditions. Participants were surveyed for a Mean Opinion Score for each scenario. Both latency and packet loss were found to negatively affect QoE, with "fast" games affected most.

Lee et al. investigated the effect of latency on thin-client games, and how different games are affected by said latency (Lee, 2012). Lee et al. chose three genres to study: action, first person shooter, and role playing, and chose three games per genre. Subjects were asked to play each game at five different latencies ranging from 0ms to 400ms. Lee et al. measured muscle movement at subjects' corrugator supercilii, a muscle near the eye that is often used when frowning or glaring, as a measure of quality of experience. First person shooter games were found to be most affected by latency, followed by role playing, then action.

Pedersen et al. examined the correlation of latency to jitter and throughput (Pederson, 2011). Jitter was measured based on variation in latency and was calculated continuously every time a ping packet was received. Jitter was based on the previous jitter value and the difference in ping times between the current and previous packets. Throughput was measured as the average maximum data rate between sender and receiver when transmitting files. The study found that there is some correlation between latency, jitter, and throughput.

2.4: Summary

Based on these previous works, thin-client games that require continuous input and higher reaction time will suffer more in terms of QoE when latency is introduced. Depending on the thin-client platform, latency may affect input delay and frame rate. Latency's impact on users' QoE depends on the type of game being played. Fast-paced games are more affected than slower-paced games.

These studies served as a basis for our hypothesis. Shea et al. and Chen et al. served as a background of how latency affects thin client games. We drew on the conclusions from Jarschel et al., Lee et al., and Pederson et al. when we designed our survey questions.

Chapter 3: Methodology

3.1: Introduction

In this chapter, we discuss the methodology of our study. Section 3.2 contains information on our hardware and software setup. We describe the physical setup of the client and server computers, the software we used to facilitate the study, and the configuration of each program. Section 3.3 contains information about our study parameters, including a discussion of the latencies we chose and the dependent variables we decided to test. Section 3.4 details the procedures followed in our study and contains our quality of experience (QoE) questions.

3.2: Experimental Setup

This section discusses the software and hardware chosen to facilitate this study as well as the reasoning behind the choices we made. Cloud games require both a server and a client to function and the choice of cloud gaming software had a big impact on subsequent design decisions. In order to test cloud gamers against artificially controlled network conditions we had to choose which cloud gaming system to use and the method for controlling latency.

3.2.1: GamingAnywhere

GamingAnywhere[3] was selected primarily because it is relatively unstudied and for its ability to work with a wide variety of games. Other thin-client services we considered were OnLive[4] and GameNow.[5] OnLive is arguably the most popular thin-client service, and many studies have been done previously related to OnLive. GameNow is similar to OnLive, but it is

---

[3] http://gaminganywhere.org/
[4] http://www.onlive.com/
[5] http://www.ugamenow.com/#/landing

smaller and its selection of games and server availability is much more limited. GamingAnywhere also has the advantage of being open-source, while the software for the other two services is locked down.

GamingAnywhere provides documentation for how to construct client and server configuration files, as well as providing a handful of configuration files they have already created. This study utilizes the Neverball configuration files provided by GamingAnywhere without any modifications.[6]

3.2.2: Neverball

The effects of latency are more pronounced in faster paced games. First person shooters were a tempting option - being a popular and well known genre (Apperley, 2006). However, this genre is biased by some players being much more skilled than others due to past experience. In addition, we determined it would be difficult to find objective ways of measuring player performance in a meaningful way.

After surveying the field, we selected the open source marble-roller Neverball, released in November 2003. The goal of Neverball is to tilt the world to get a marble around obstacles and to a marked goal. The only controls in Neverball are the arrow keys. The simple arrow key controls and relatively fast paced gameplay allowed for the average test participant to pick up the game quickly. In addition, there is a time limit to complete the level. This provides an objective measure of player performance built right into the game.

After choosing Neverball, we found that many of the more challenging Neverball levels were too hard to reasonably expect participants who lacked prior experience to complete the level. On the other hand, the easiest levels are so easy that the changes in latency we planned to

[6] http://gaminganywhere.org/dl/config/server.neverball.conf

measure would not have a significant effect on each player’s performance. It would be unfortunate to lose the significance of the data because of insufficient granularity in player performance.

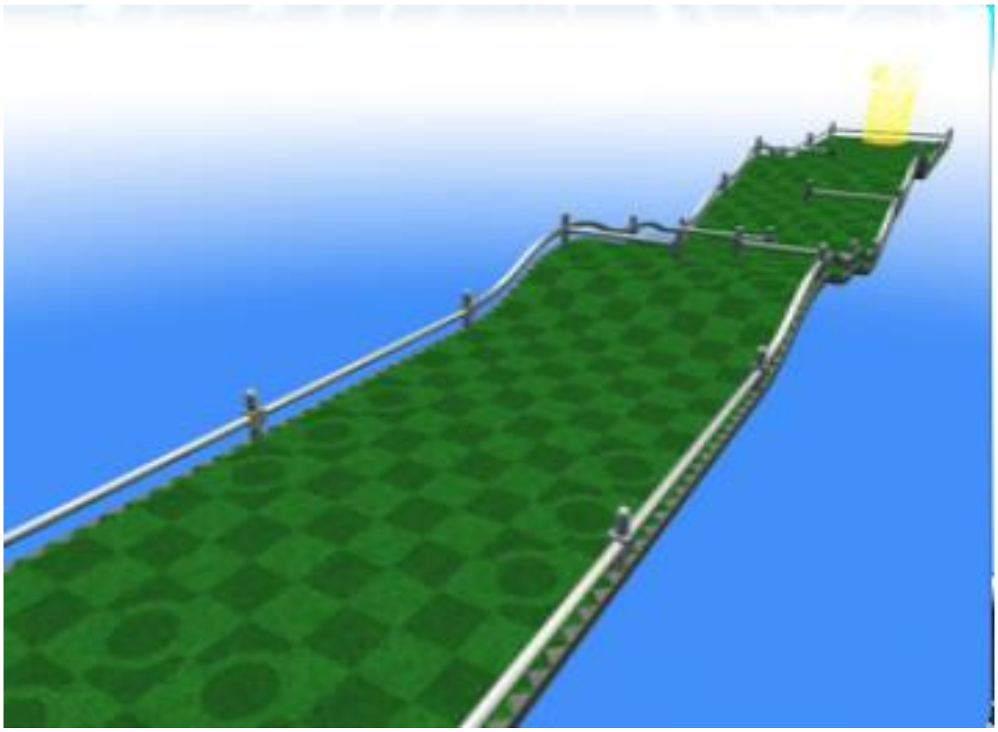


Figure 3.1 Level 7 in Neverball, chosen for our study.

After much deliberation and many test-runs of different levels, we chose level seven from the “Easy” Neverball level set (Figure 3.1). This level is composed of a hill-and-valley curved ramp with maze-like obstacles on its surface. The platform is completely walled-in, making it difficult for the ball to fall off of the world (Figure 3.2). The level must be completed in ninety seconds or the player will time-out. Similarly, if the player maneuvers the ball off of the map the player gets a “fall out.” Initial test runs indicated that most people would be able to complete the level with a little time to spare.

Figure 3.2 Play view of Neverball (http://neverball.org/screenshots.php?id=07&set=01)

3.2.3: Dummynet

Latency between the client and server was controlled with the program Dummynet.[7] Dummynet was installed on and manipulated from the server computer. This served to keep the participants oblivious to the exact nature of the network setup. We wrote a short script to automate the process of setting up the appropriate Dummynet rules and pipes to create the desired latency at any input level (provided in the Appendix 1.1).

3.2.4: Hardware Setup

The server needed to multitask running Neverball, the GamingAnywhere server and Dummynet simultaneously. None of these programs are too resource-intensive, and in the end a laptop with an Intel i7 processor, Nvidia GeForce graphics card and 16 gigabytes of RAM was

[7] http://info.iet.unipi.it/~luigi/dummynet/

used to host the game. More exact specifications are included the Appendix 2.4. This machine was most suitable both because its hardware was supported by GamingAnywhere and also because it had sufficient resources and power to operate smoothly in testing.

Surprisingly this same machine was not suitable as a client (its originally intended role) because GamingAnywhere does not support hardware rendering using the computer's graphics card. The GamingAnywhere client's software renderer was found to have insufficient performance to play Neverball in real time. An Intel Desktop with an i7 processor and Intel onboard graphics was chosen to be the client because its hardware was supported by the GamingAnywhere renderer. The server and client computers were connected by an Ethernet cable.

3.2.5: Benchmarking

Benchmarking was conducted using the Fraps utility.[8] Fraps is free software that can measure frame rate on a given running process. Additionally, GamingAnywhere exposes the encoding frame rate on the server and the decoding frame rate on the client. The server was encoding at fifty frames per second and the client lost only one or two of those frames. The setup performance degraded at higher resolutions, so a low resolution (600x800) was selected.

3.3: Study Parameters

Our study was designed to test participants' performance and subjective measures of experience against various levels of network latency. Our study sought to identify a hypothesized relation between network latency, player performance, and quality of experience. As such, the

[8] http://www.fraps.com/

independent variable in our study was the network latency between the client and server computers. The dependent variables measured were the time it took participants to complete the level and the quality of experience of the participants.

In order to achieve a controlled environment, each participant was tested individually with two researchers in the room. One researcher operated the server and changed the network latency between trials. The other researcher explained the procedures to the participant, recorded data, and answered the participant's questions.

3.3.1: Latencies

In our pilot studies, we performed mock trials to test the connection between server and client, chose the network latencies to test participants with, and prepared our study procedures. During these pilot studies, we determined the maximum network latency we should test participants at was 300ms based on our ability to complete the chosen level. Later, beta tests prompted us to lower that maximum network latency to 200ms as the beta testers could not finish the level at the higher latencies. To have a comprehensive dataset, five latencies were chosen to test based on this result: 33ms, 66ms, 100ms, 150ms, and 200ms. Each trial in Neverball was played at one of these network latencies. A network latency of 0ms was also tested. It was determined, however, that a network latency of 0ms is unrealistic in a real-world thin-client scenario. The 0ms test, which played as if the game were local to the client computer, was used only as a practice round for participants, but still provided useful information on a best-case scenario.

In order to reduce the effects of participant improvement between trials on the data, two additional aspects were introduced to our study. As mentioned earlier, players were given a practice round at 0ms in order for them to learn the basics. Additionally, the order in which the network latencies were given after the practice round was scrambled based on how many participants had gone previously. The latencies were divided into three groups: 33ms and 66ms, 100ms, and 150ms and 200ms. Each test participant played the level with each network latency in order of one of the six combinations of these groups. For instance, if one participant played the trials in the order 33ms, 66ms, 100ms, 150ms, and 200ms, the next participant would play the trials in the order 33ms, 66ms, 150ms, 200ms, and 100ms.

3.4: Study Procedures

At the beginning of each session, each participant was read a script detailing instructions for the study. The script is included in the Appendix 2.1 along with our formal step-by-step instructions for the study. Participants were then prompted to enter demographic information including their age, gender, and previous videogame experience. Once finished, participants were given six trials to complete the chosen level in Neverball, each with a different latency at a preset random. Every participant was given a practice trial at 0ms first, then a randomized order of the other latencies. For each trial, the participant’s time to complete the level was recorded using Neverball’s in-game timer along with a few subjective questions about the trial. Each question was answered on a ascending scale of one to five. The questions we asked were:

1. How would you rate your overall quality of experience?
2. How would you rate your performance?
3. How responsive were the controls when you were playing the game?
4. How much do you think your performance was affected by the responsiveness of the controls?

5. How enjoyable was the game?

In addition to the script, we had a list of answers to frequently asked questions. Most of the questions we came up with ourselves, but we added to the list after running our beta tests. The main addition after the beta tests was to ask the participants to try to avoid falling off the level, as fall-out data was not useful to us.

Participants were gathered from the university campus through a mass email and by word of mouth. To add incentive, participants were given the opportunity to enter a raffle for a Newegg.com gift card for $75. Every participant entered the raffle.

3.5: Summary

We conducted our study using a thin-client setup with two computers, a client and server, connected via Ethernet cable. The server computer ran Neverball through GamingAnywhere using Dummynet to control latency. The client computer received a stream from the server computer. We tested participants' performance and QoE against changes in latency between the server and client. Participants played the same level at latencies 33, 66, 100, 150, and 200 ms, and a practice round at 0 ms. We recorded participants' time to complete the level at each trial to measure performance. Participants also answered 5 subjective questions after every trial.

Chapter 4: Analysis

4.1: Introduction

This section documents the statistical analysis that was performed on the gathered data as well as the conclusions we drew from the analysis. The first subsection details the statistical analysis, both in terms of means-testing as well as visualizations. IBM's Statistical Package for the Social Sciences (SPSS)[9] was utilized to conduct the statistical analysis and to produce the graphs. The next subsection discusses the demographics and associated trends in our data. Finally, the last subsection is a conclusion based on the discussion in the prior subsections.

We have summarized our results and analysis below in brief and cover the same in detail in the subsequent subsections. We used two different types of means testing: the Friedman test for the discrete survey response data and a within-subject one-way analysis of variance for the continuous performance data (Urdan, 2010 and Dueker, 2012). The null hypotheses, across data types, were always that the players' means were equivalent across latencies. Our alternative hypotheses represent those cases where the means differ in a statistically significant way. These could be written as follows:

$$H_0 := \mu_{33} = \mu_{66} = \mu_{100} = \mu_{150} = \mu_{200}$$

$$H_a := \mu_{33} \neq \mu_{66} \neq \mu_{100} \neq \mu_{150} \neq \mu_{200}$$

Where H_0 and H_a are the null and alternative hypothesis respectively and μ_n is the mean for the trial at latency level n.

The repeated measures analysis of variance (RANOVA) and the Friedman Test both compute a p-value which represents the probability of having observed the given data under the null hypothesis. The RANOVA is a statistical method used to test hypotheses when the same

[9] http://www-01.ibm.com/software/analytics/spss/

group of subjects is repeatedly tested at different levels of some factor. The Friedman Test is an analog to RANOVA that allowed us to test the survey data. We chose a statistical significance or alpha level of 0.05. Comparison of the computed p-value for a given data set with the chosen level of significance indicates whether or not the null hypothesis should be rejected. We rejected the null hypothesis only when the p-value was less than the chosen alpha level and failed to reject it otherwise. The necessities that under laid our reasoning for the chosen methods are explained in detail in subsequent sections. A brief overview of our results follows, first for the performance data and then for the survey response data.

We measured players' performances by the time it took them to complete the level. This data is continuous and measured in seconds between zero and ninety, inclusive. Players who failed to complete the level were treated as having taken the maximum time to complete the level or ninety seconds.

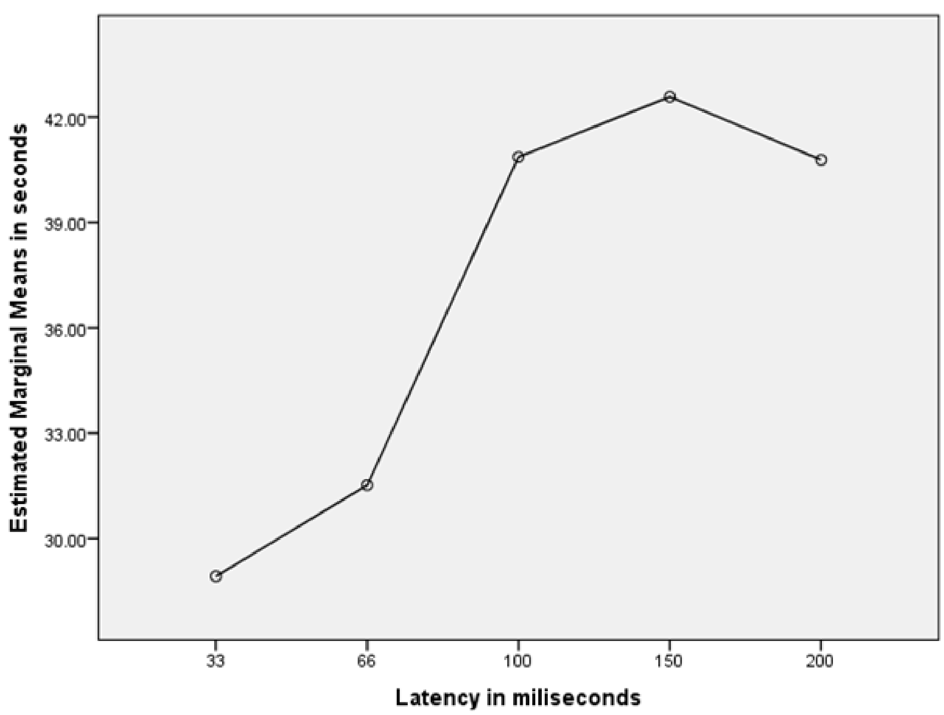


Figure 4.1: A graph of the average time to complete at each latency.

The marginal means plot (Figure 4.1) provided above conveys the general trend. SPSS supplied appropriate corrections to the data because of its observed imperfections. An RANOVA utilizing the Huynh-Feldt correction determined that the mean player time-to-complete varied statistically significantly between trials ($F(3.527,116.401) = 5.362, p=.001$) (Rutherford, 2012). Post hoc testing with a Bonferroni correction revealed that increased latency resulted in an increased mean time to complete (Rutherford, 2012). The Huynh-Feldt correction was applied to account for the observed lack of homogeneity of the variances of the differences between all possible pairs of groups. The Bonferroni correction allows for the comparison of means between samples that are not independent, as is the case in repeated measures design. There were statistically significant increases in player time to complete between trials at 66 ms to 100 ms, and at 100ms to 150 ms. There was no evidence of statistically significant increases in player time to complete between trials at 33ms and 66ms as well as between 150ms and 200ms. Therefore, we conclude that increased latency resulted in decreased player performance (or increased time to complete) but not when latency was less than 66ms and not when latency was greater than 150ms.

The player survey response data was collected after each trial. Players reflected on their gameplay and answered various subjective questions related to performance and QOE. This data was collected in discrete values between one and five inclusive. There was a statistically significant difference in player perceived quality of experience between trials, $\chi^2(4) = 17.280$, $p = 0.002$.[10] There was a statistically significant difference in player perceived performance, $\chi^2(4) = 18.277$, $p = 0.001$. There was no statistically significant difference in player perceived responsiveness between trials, $\chi^2(4) = 8.297$, $p = 0.08$. There was no statistically significant

[10] The Chi-Squared distribution with K degrees of freedom represents distribution of the sum of squares for K variables (Rutherford, 2012)

difference in player perceived effect of responsiveness on performance, $\chi^2(4) = 1.033$, $p = 1$.

There was no statistically significant difference in player perceived enjoyability between trials, $\chi^2(4) = 5.565$, $p = .234$.

4.2: Statistical Analysis

In our study we sought to analyze various dependent variables, including player performance and quality of experience, against the independent and controlled latency variable. Because we tested the same subjects repeatedly, ours was a repeated-measures or within-subject type of study. Once data was collected, we used means-testing to see if there was a statistically significant difference in means of the various dependent variables for each player. The data fits into two categories: objective player performance measured in time to complete the level, and subjective player responses about quality of experience measured from one to five. We used box and whisker charts to identify the basic trends between each treatment level. We have divided the analysis between the performance and survey data because the different kinds of data required different techniques. Our chosen alpha or level of significance was 0.05.

4.2.1 Performance Analysis

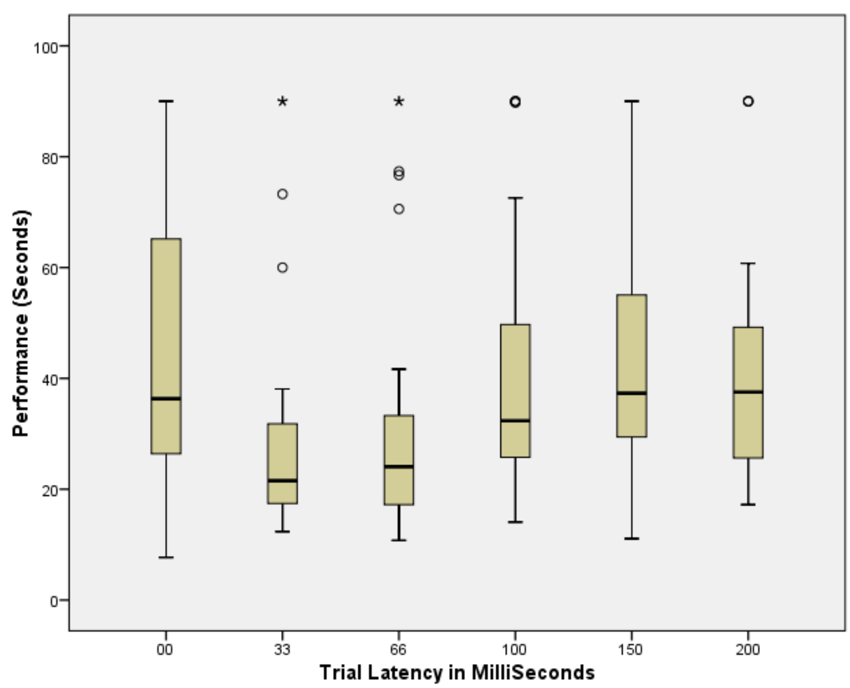


Figure 4.2: Box-and-Whiskers plot on the Player Performance data.

Figure 4.2 shows the distribution of player time-to-complete the level for each latency level. The central line is the median of that group, and the upper and lower bounds are the 3rd and 1st quartiles respectively. The whiskers show 1.5 times the interquartile range (IQR). The IQR is equivalent to the difference in the 1st and 3rd quartiles or the middle fifty percent of the data. Circles represent outliers or data points beyond 1.5 times the IQR. Stars represent extreme outliers or data points beyond 3 times the IQR. This box chart shows that the medians trend generally upward for increasing latency.

The trial round at 0ms latency exhibited significantly more variability (Figure 4.3). We examined the variances of the six trials further in tabular form:

0ms	33ms	66ms	100ms	150ms	200ms
679.959	401.656	497.634	512.463	435.462	383.330

Figure 4.3: Variance of the six trials

We took the larger variance for the practice round as good evidence that it should be discarded from further analysis. This round was designed for players to learn the game and their learning was seen in this increased variance.

We performed a repeated measures analysis of variance for the five trials at 33, 66, 100, 150 and 200 milliseconds. The assumptions for repeated measures ANOVA had to be considered before analysis could commence:

1. Dependent data must be continuous.
2. Data must be in matched groups.
3. Variances of dependent data groups must be similar.
4. Distribution of the dependent variable must be approximately normal.
5. Data must meet Sphericity assumption.

The first condition requires that the dependent variable be measured on a continuous scale between two points, with the possibility of taking any value between. The performance data was continuous because we recorded player performance as any and all possible value between 0 and 90 seconds so this first assumption was easily met. The second condition was met by our study design. We matched groups of 34 people for each of the levels of latency. The third condition requires homogeneity of variances across the groups. The variances for each of the latency levels in our study are listed above and for the five trials of interest the variances are homogeneous. The fourth requirement, that the data be approximately normal, was satisfied as well. Figures 4.4-4.8 provide Quantile-Quantile plots that show that the performance data is approximately normal. Quantile-Quantile (q-q) plots compare a given distribution against the standard normal distribution. We have provided q-q plots for each of the five levels we analyzed. The assumption that all of the variances of the differences between trials are homogenous is called sphericity by statisticians (Rutherford, 2012). We tested this assumption in SPSS using Mauchly's Sphericity Test (Everitt, 2002). Below the SPSS output is provided, with annotations.

We first used SPSS to generate q-q plots for our data so we could test the normality assumption. Ideal data would fall perfectly in line with the trend line in each plot. While not perfect, our data was approximately normal as shown below in Figures 4.4 through 4.8. Analysis of Figures 4.4-4.8 reveals plots of datapoints that closely follow the trendline, indicating that the data is nearly normal. Some groupings are better than others, and the earlier plots for the 33 and 66 trials show significant outliers.

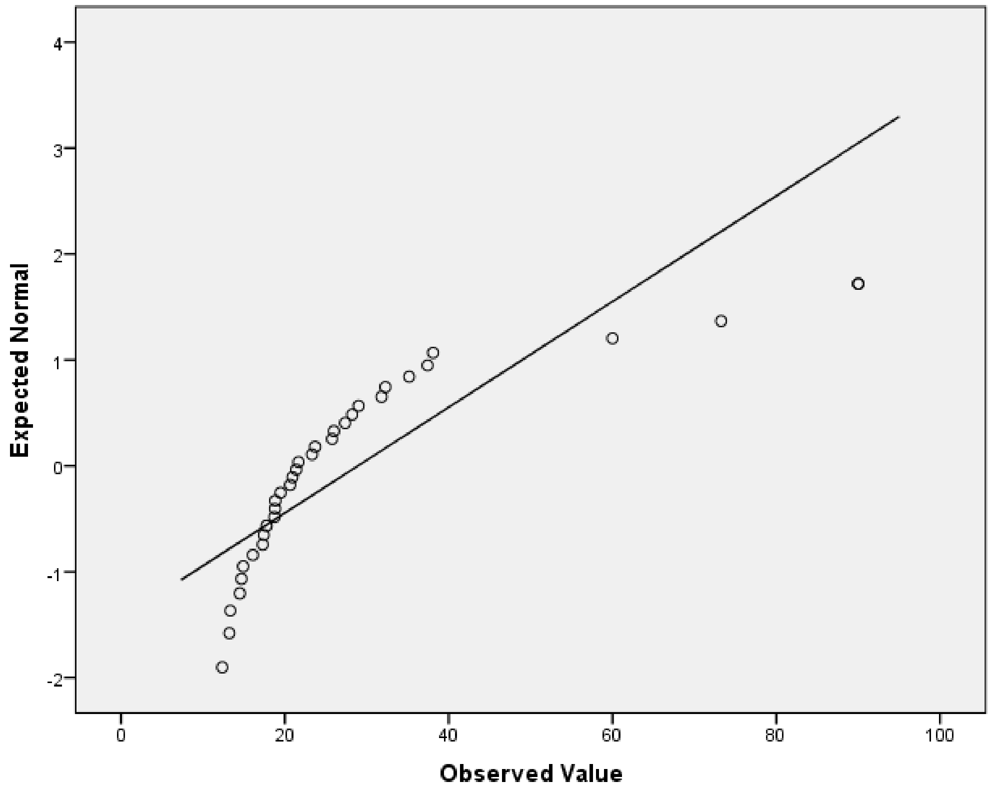


Figure 4.4: A normal quantile-quantile plot of latency 33ms.

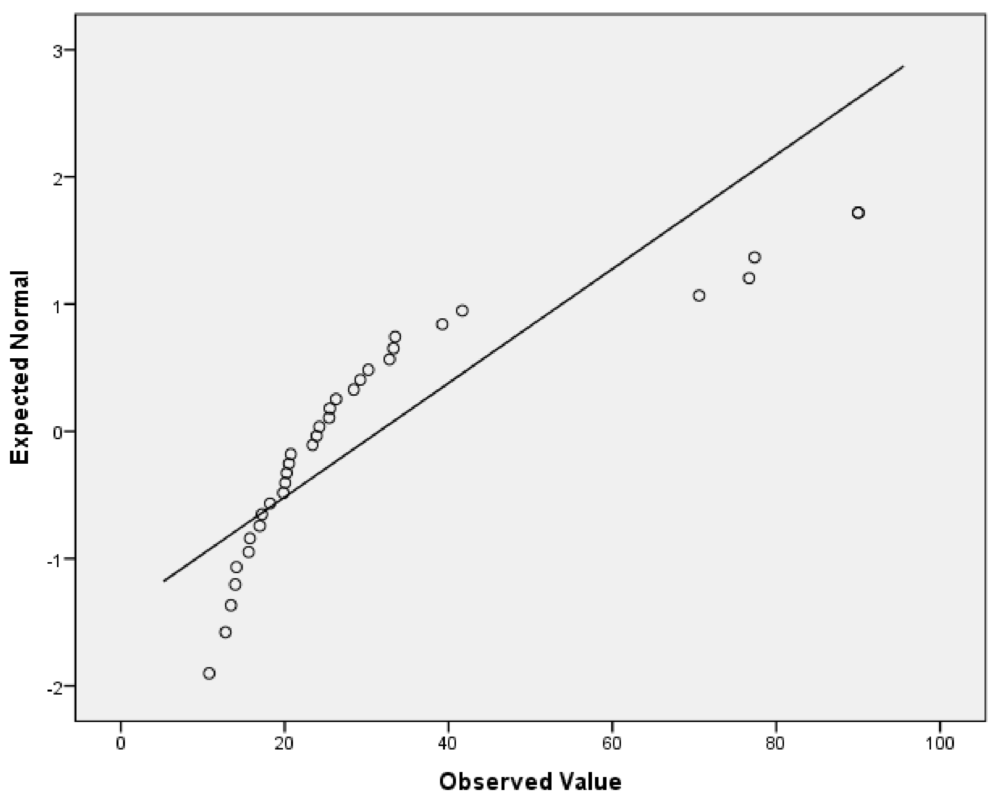


Figure 4.5: A normal quantile-quantile plot of latency 66ms.

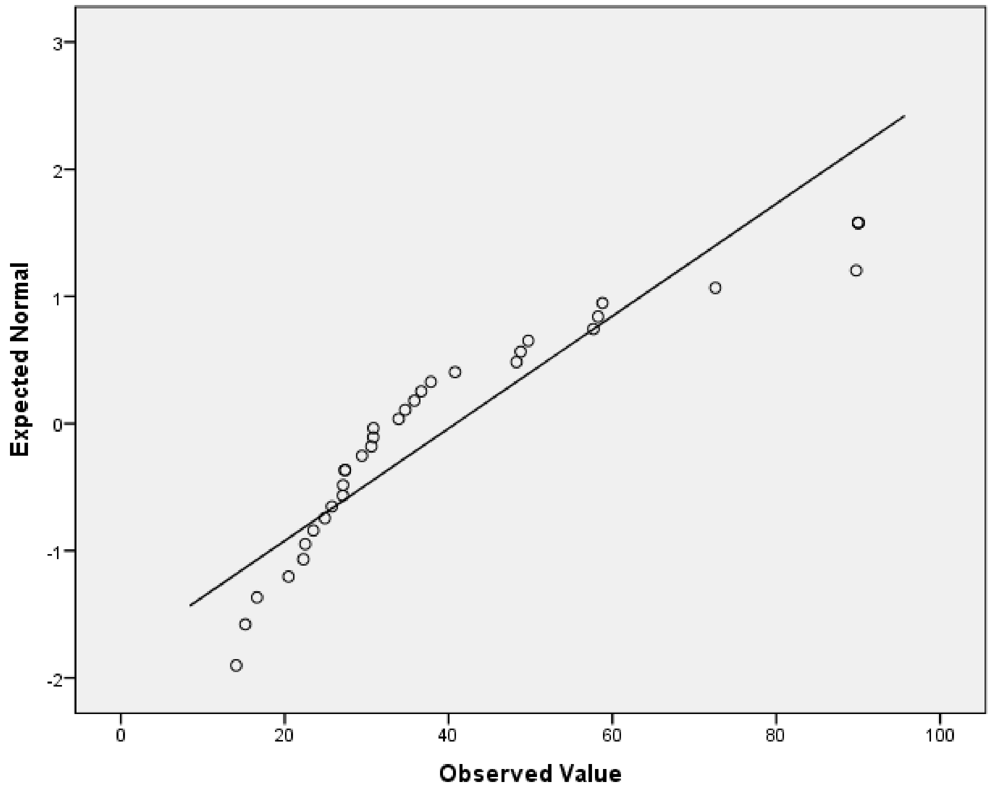


Figure 4.6: A normal quantile-quantile plot of latency 100ms.

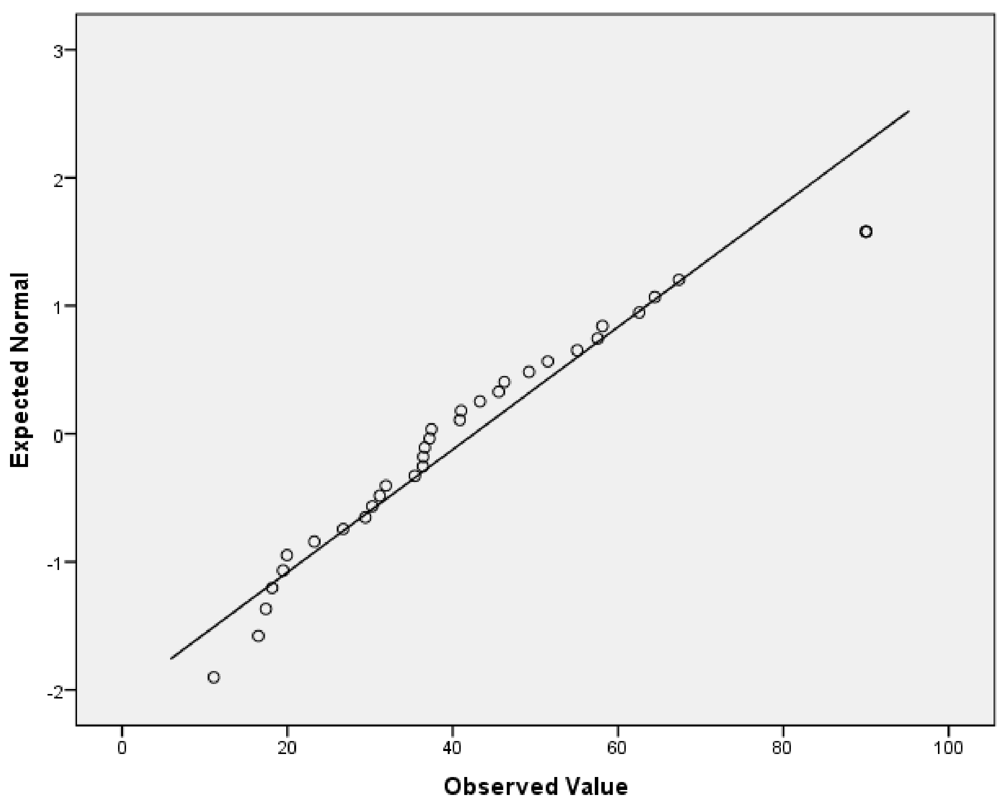


Figure 4.7: A normal quantile-quantile plot of latency 150ms.

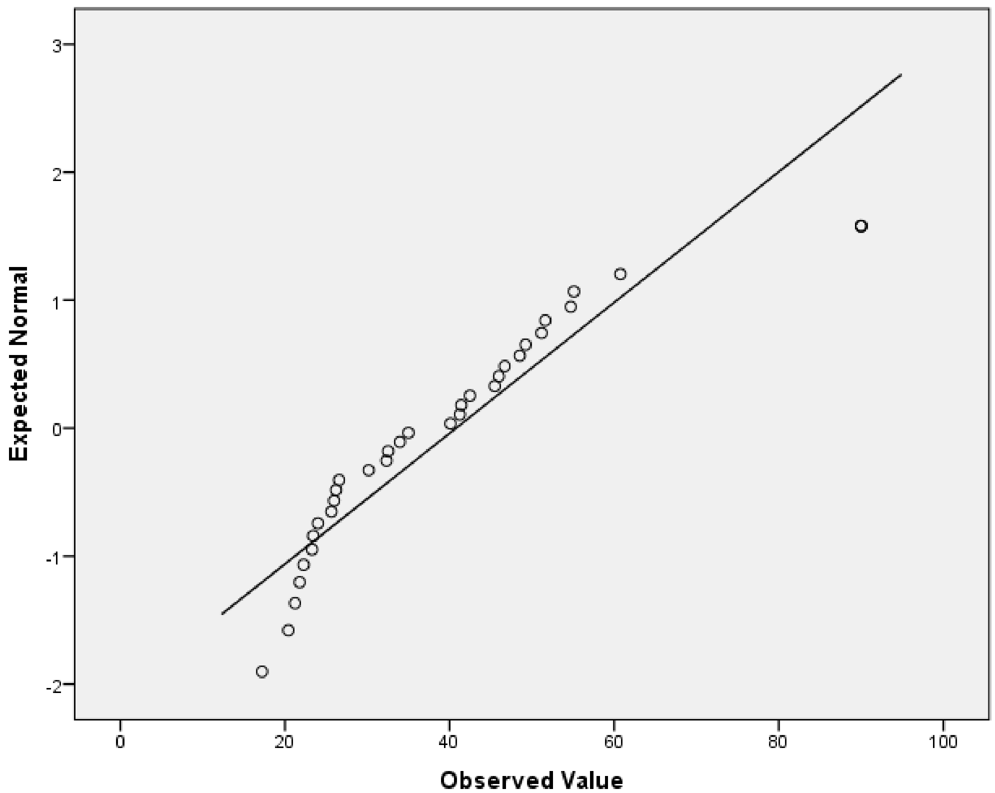


Figure 4.8: A normal quantile-quantile plot of latency 200ms.

After checking the normality assumption, we used SPSS to conduct the repeated measures analysis of variance. SPSS first conducted Mauchly's test of Sphericity to test that assumption. Figure 4.9 shows the descriptive statistics for the performance data and the results of Mauchly's test.

Descriptive Statistics			
	Mean	Std. Deviation	N
33 ms	28.9235	20.04135	34
66 ms	31.5197	22.30771	34
100 ms	40.8615	22.63764	34
150 ms	42.5762	20.86773	34
200 ms	40.7803	19.57883	34

Mauchly's Test of Sphericity[a]						
Within Subjects Effect	Mauchly's W	Approx. Chi-Square	df	Sig.	Huynh-Feldt[b]	Lower-Bound[b]
Latency	.498	21.908	9	.009	.882	.250

Figure 4.9: Tests the null hypothesis that the error covariance matrix of the orthonormalized transformed dependent variables is proportional to an identity matrix.[a]

a. Design: Intercept
Within Subjects Design: Latency

b. May be used to adjust the degrees of freedom for the averaged tests of significance. Corrected tests are displayed in the Tests of Within-Subjects Effects table.

The results of Mauchly's Test of Sphericity indicate that the assumption of sphericity has been violated because $p < .05$. The Mauchly's W statistic indicates that sphericity is a problem because it does not equal one (Everitt, 2002). The null hypothesis for this test is that the data is spherical and the alternative hypothesis is that the data is not spherical. With the computed p-value of .009 for the performance data, we reject the null hypothesis and accept the alternative - the data is not spherical.

SPSS provided suitable corrections because of the observed departure from sphericity. These take the form of Epsilon values, which represent the severity of the departure from sphericity from 0 to 1 with 1 being no departure. Epsilon is used to adjust the degrees of freedom for the analysis, which results in corrected p-values. We chose the Huynh-Feldt correction because it was the least conservative and the observed departure from sphericity was not severe (Rutherford, 2012). The lower-bounds correction is conservative but nevertheless a good lower bounds for the purposes of comparison because it represents the worst case. The lower bounds estimate is simply calculated as $1 / (1 – k)$ where K is the number of repeated measures. SPSS provided the results of the analysis under both of these corrections as well as with sphericity assumed.

Tests of Within-Subjects Effects							
Measure: TimeToComplete							
Source		Type III Sum of Squares	Df	Mean Square	F	Sig.	Partial Eta Squared
Latency	Sphericity Assumed	5288.202	4	1322.051	5.365	.000	.140
	Huynh-Feldt	5288.202	3.527	1499.225	5.365	.001	.140
	Lower-bound	5288.202	1.000	5288.202	5.365	.027	.140
Error(Latency)	Sphericity Assumed	32526.117	132	246.410			
	Huynh-Feldt	32526.117	116.401	279.433			
	Lower-bound	32526.117	33.000	985.640			

Figure 4.10 shows the results of our within-subject analysis of variance.

Figure 4.10 shows the results of the analysis. SPSS provided a significance value for each different correction that was applied. The repeated measures analysis of variance tests an omnibus null hypothesis (H_0) against an alternative (H_a) as such:

$$H_0 := \mu_{33} = \mu_{66} = \mu_{100} = \mu_{150} = \mu_{200}$$

$$H_a := \mu_{33} \neq \mu_{66} \neq \mu_{100} \neq \mu_{150} \neq \mu_{200}$$

The null hypothesis represents the case where all the mean player performance at each latency level is the same and the alternative representing the case where they differ.

The results were significant and reveal an elicited trend between latency and player performance. The lower-bound estimate was very conservative but the results were still significant at $p = .027$. Because we had significant results in the first part of the analysis, SPSS continued and performed post-hoc pairwise comparisons. These comparisons were used to see if there was a statistically significant difference between any given pairing and in what direction. The Bonferroni correction is used. This correction allows pairwise comparisons between groups that are not independent, as in repeated measures designs. The Bonferroni correction adjusts our alpha or significance level by the number of repeated measures. For these pairwise comparisons the alpha level is .05/5 = .01.

Estimates				
Measure: Time To Complete				
Latency	Mean	Std. Error	95% Confidence Interval	
			Lower Bound	Upper Bound
1	28.924	3.437	21.931	35.916
2	31.520	3.826	23.736	39.303
3	40.861	3.882	32.963	48.760
4	42.576	3.579	35.295	49.857
5	40.780	3.358	33.949	47.612

Figure 4.11: Estimated means of players' time to complete the level.

Pairwise Comparisons						
Measure: Time To Complete						
(I) Latency	(J) Latency	Mean Difference (I-J)	Std. Error	Sig.	95% Confidence Interval for Difference	
					Lower Bound	Upper Bound
1	2	-2.596	3.960	1.000	-14.509	9.316
	3	-11.938*	3.337	.011	-21.976	-1.900
	4	-13.653*	4.246	.029	-26.426	-.880
	5	-11.857*	3.626	.025	-22.764	-.949
2	1	2.596	3.960	1.000	-9.316	14.509
	3	-9.342*	2.936	.032	-18.174	-.510
	4	-11.056*	3.079	.011	-20.320	-1.793
	5	-9.261	4.516	.483	-22.847	4.326
3	1	11.938*	3.337	.011	1.900	21.976
	2	9.342*	2.936	.032	.510	18.174
	4	-1.715	4.106	1.000	-14.067	10.638
	5	.081	4.047	1.000	-12.094	12.256
4	1	13.653*	4.246	.029	.880	26.426
	2	11.056*	3.079	.011	1.793	20.320
	3	1.715	4.106	1.000	-10.638	14.067
	5	1.796	3.901	1.000	-9.939	13.530
5	1	11.857*	3.626	.025	.949	22.764
	2	9.261	4.516	.483	-4.326	22.847
	3	-.081	4.047	1.000	-12.256	12.094
	4	-1.796	3.901	1.000	-13.530	9.939

Figure 4.12: Pairwise Comparisons for player performance based on estimated marginal means
* The mean difference is significant at the .05 level.

Figure 4.12 reveals that the observed trends in the performance data was supported by means testing. There is a row for every pairing of treatments (I,J) where I, J are integers 1 through 5 to represent the five trials at 33, 66, 100, 150, and 200 milliseconds respectively. The mean difference column displays the difference between the means. Differences that are statistically significant are marked with an asterisk. The table provided above showed that most of the lower treatment means were statistically significantly lower than the later groups. However, not all pairwise combinations were statistically significant. We observe that the 33 and 66 pair and the 150 and 200 pair did not yield significant results. This indicates that the players were not as impacted by the changes between these treatments as the others.

The pairwise comparisons show that players' performance began to decrease at latencies above 66ms, but did not decrease further past 150ms. Players' performance was not significantly affected below 66ms. We have identified the bars below and above which players are not affected by changes in latency. Overall, the repeated measures analysis of variance yielded results in line with our hypothesis that player performance and quality of experience is directly related to latency in cloud games.

4.2.2 Survey Response Analysis

We could not use the repeated measures ANOVA for the survey response data. The survey response data was discrete, not continuous in nature. Since answers occur in integer ranks of one through five it cannot take on all values between one and five and is therefore discrete. This violates the assumptions of repeated measures ANOVA and so we used the Friedman Test instead.

We used nonparametric testing to do the means testing for the survey response data. We used the Friedman Test, which unlike repeated measures ANOVA does not make assumptions about the distributions of the dependent data. We performed the Friedman Test for each of the five different questions that were asked on the survey.

The Friedman Test uses a null hypothesis (H_0) that the means for every trial are equal for our study. This means there is no correlation between latency and the mean of the five survey response questions at each latency (μ_n). The alternative hypothesis (H_a) represents the condition that the means are different. This means latency has a significant effect on the mean of the five survey response questions at each latency (μ_n).

$$H_0 := \mu_{33} = \mu_{66} = \mu_{100} = \mu_{150} = \mu_{200}$$

$$H_a := \mu_{33} \neq \mu_{66} \neq \mu_{100} \neq \mu_{150} \neq \mu_{200}$$

Once again, we chose an alpha level of 0.05. The results have been recorded below in Figures 4.13 through 4.17.

4.2.3 Friedman Tests

"How would you rate your overall quality of experience?"

Ranks	Mean Rank
33 ms	3.40
66 ms	3.31
100 ms	3.26
150 ms	2.51
200 ms	2.51
Test Statistics	
N	34
Chi-Square	17.280
df	4
Asymp. Sig.	.002

Figure 4.13: Friedman test on QoE.

"How would you rate your performance?"

Ranks	Mean Rank
33 ms	3.56
66 ms	3.54
100 ms	2.75
150 ms	2.40
200 ms	2.75
Test Statistics	
N	34
Chi-Square	18.277
df	4
Asymp. Sig.	.001

Figure 4.14: Friedman test on subjective player performance.

"How responsive were the controls when you were playing the game?"

Ranks	Mean Rank
33 ms	3.24
66 ms	3.19
100 ms	3.24
150 ms	2.60
200 ms	2.74
Test Statistics	
N	34
Chi-Square	8.297
df	4
Asymp. Sig.	.081

Figure 4.15: Friedman test on subjective responsiveness.

"How much do you think your performance was affected by the responsiveness of the controls?"

Ranks	Mean Rank
33 ms	2.88
66 ms	2.91
100 ms	3.13
150 ms	3.07
200 ms	3.00
Test Statistics	
N	34
Chi-Square	1.033
df	4
Asymp. Sig.	.905

Figure 4.16: Friedman test on responsiveness effect on performance.

"How enjoyable was the game?"

Ranks	Mean Rank
33 ms	3.16
66 ms	3.09
100 ms	3.25
150 ms	2.75
200 ms	2.75
Test Statistics	
N	34
Chi-Square	5.565
df	4
Asymp. Sig.	.234

Figure 4.17: Friedman test on game enjoyability.

The results of the various Friedman Tests were mixed. Some questions yielded statistically significant results and some did not. The questions on Quality of Experience and subjective performance were statistically significant at $p = .002$ and $p = .001$ respectively but the other questions did not. For both of these questions, the mean rank decreased as latency increased with a sharp decrease between 66 and 100 ms. Taking into account the results of the Friedman test and the observed means, we conclude that QoE and subjective performance both decreased as latency increased. The analysis of the subjective survey response data resulted in strong support of our initial hypothesized relationship between latency and player reported Quality of Experience.

4.3: Demographics

Before running through the level, participants were asked to answer some demographic questions. In addition to age and gender, we asked participants how much experience they had with video games on a scale of one to five. We also asked what types of consoles and genres of games the participants had experience with. After the participants finished with the study, we asked them if they had played a game similar to Neverball in the past. We also asked why they decided to participate in the study.

The first demographic question we asked participants was: "On a scale of 1-5 (5 being the highest), how much past experience do you have with video games?" All 34 participants answered between 3 and 5. Two participants answered 3, eight answered 4, and twenty answered 5. Four participants did not answer the question (and their data has been omitted for the purposes of this question). Figure 4.18 is a graph of the performance of each of these three groups.

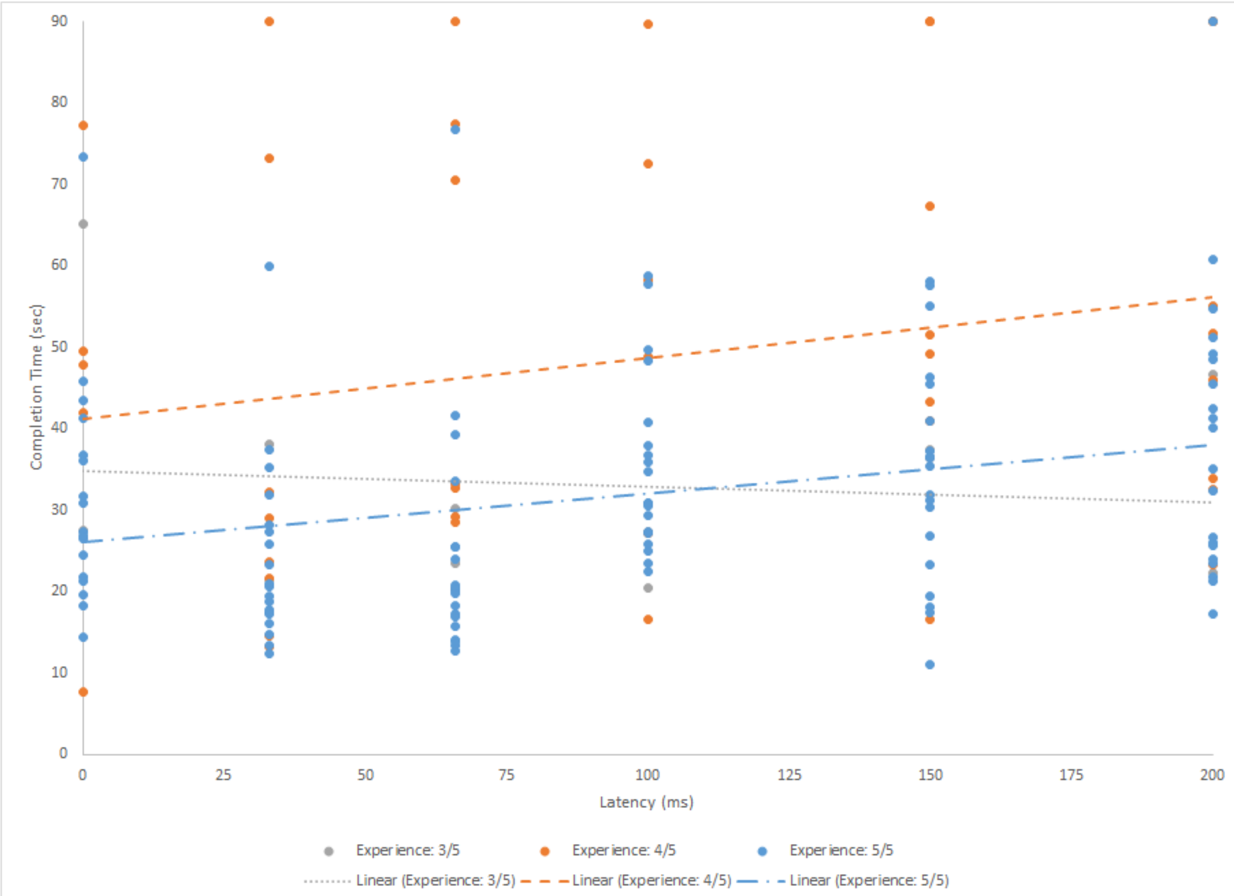


Figure 4.18 A graph of the performance of different experience groups.

While the trendline for participants who answered "3" is interesting, only two participants are considered in that group, so no solid conclusions can really be drawn from that data. It is clear from the graph however that the participants who answered "5" completed the level much faster than the participants who answered "4" on average. The two trendlines have very similar slopes (both of which match the slope of the data as a whole), showing that the different groups were not any better or worse than average at different latencies.

The next question we asked participants was: "What consoles do you use regularly?" We listed checkboxes for XBox consoles, Playstation consoles, Nintendo consoles, handheld consoles, computer, mobile, and other. The "other" field was rarely used, and nearly every participant checked off "computer" as one of their answers, so that data was not used. For each of the other types of consoles, a performance graph of participants who use the console and do not use the console was created. As an example, the graph for hand-held devices is Figure 4.19 below.

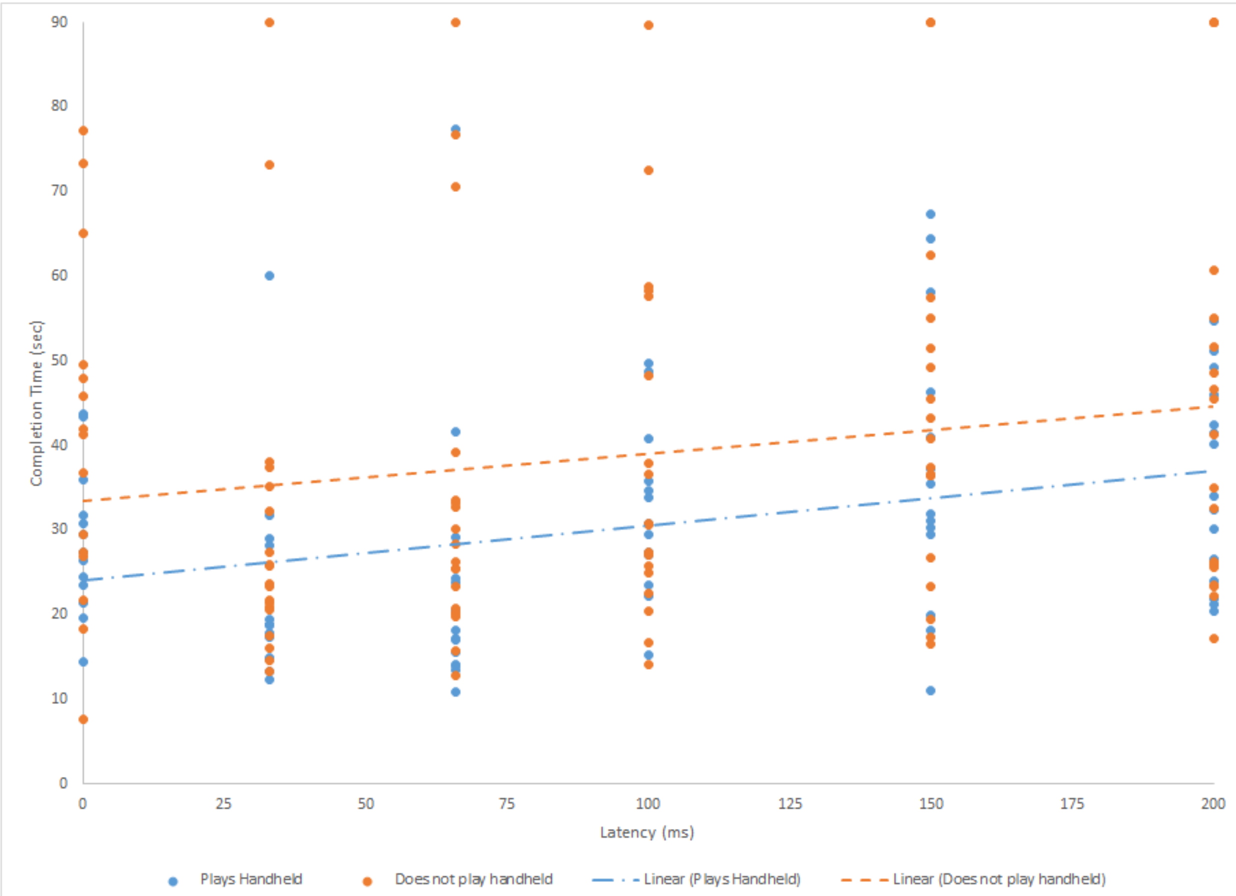


Figure 4.19 A graph of the performance of players of hand-held games.

The graph shows a ten-second difference between the average completion time of participants who use hand-held games and those who do not. Using this data, it is possible to argue that individuals who have experience with hand-held games are better at playing the game chosen for this study: Neverball. The graph of players of mobile devices is very similar. The graphs for XBox, Playstation, and Nintendo consoles all show that participants who use the consoles have no advantage on average over participants who do not use the consoles. The graphs of all of the console data are included in Appendix 4.1. There is no graph of the computer data because 33 of the 34 participants stated that they have experience with computer games. Similar to the console question, we also asked the participants what genres of games they have experience with. Unfortunately, there was not enough variability in the responses for the graphs to reflect actual findings.

The age range of the study participants was 18-22. We broke this up into two groups, and graphed the performance data just as for the previous questions. The data shows that the two age groups did not have an advantage in either direction on average. For the gender graph, only three of the participants were female, so conclusions cannot be accurately drawn from it. The graphs of the age and gender data are included in Appendix 4.1.

One final question was asked of every participant at the end of the study. The participants responded to whether or not they had previously played a game similar to Neverball. The two groups were evenly split, and the graph of the data is Figure 4.20.

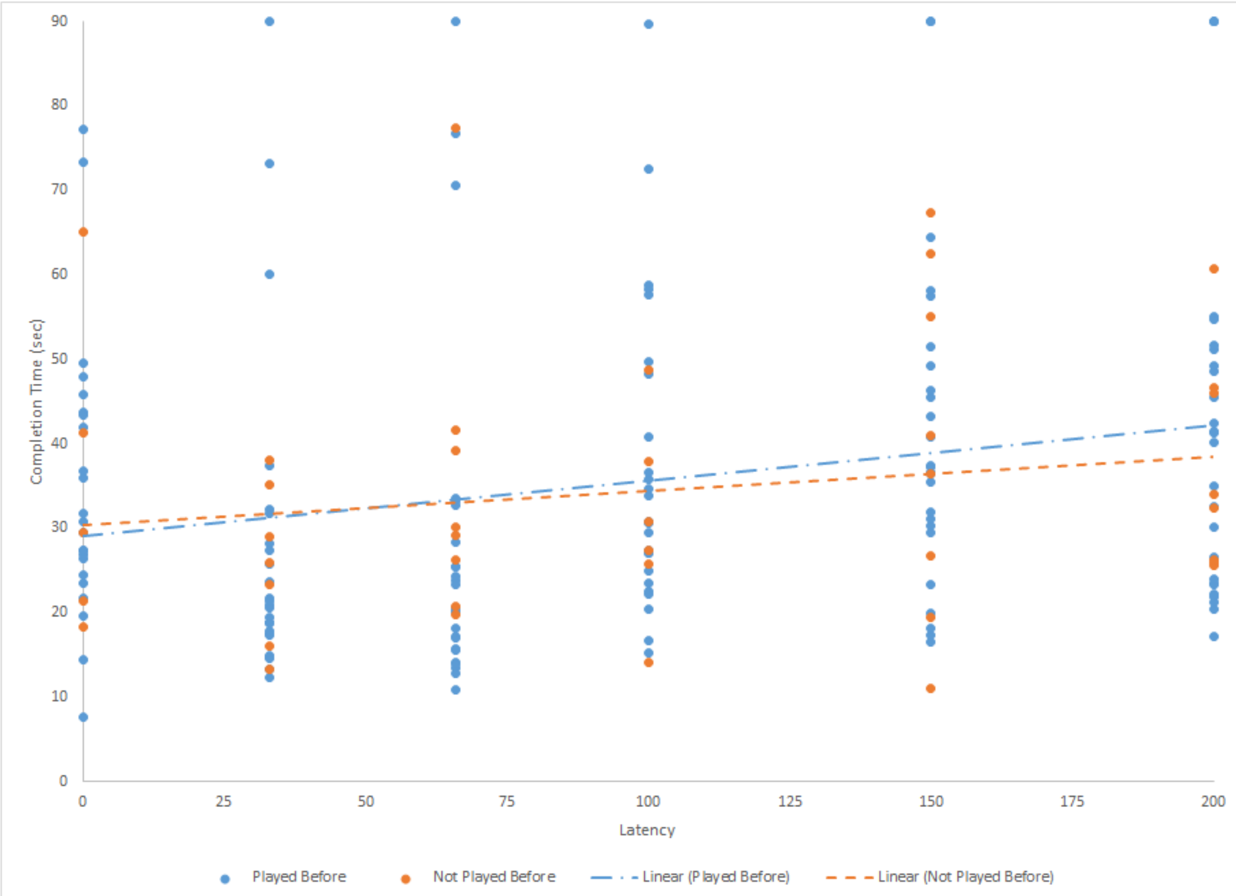


Figure 4.20 A graph of the performance of participants with and without previous experience with Neverball-like games.

Figure 4.20 shows that participants who had experience with Neverball-like games did not have an advantage on average over participants without said experience. This information is reassuring for us, as it supports that our data overall may not have been affected by player improvement between rounds.

4.4: Summary

We used a within-subject one-way analysis of variance means test to analyze the performance data, and the Friedman test to analyze the survey response data. Our repeated measures ANOVA test determined that the mean player performance varied significantly between trials (F(3.527,116.401) = 5.362,p=.001). Our player performance analysis determined the mean players' performance decreased as network latency increased. There was a sharp decrease in mean players' performance between 66ms and 100ms. There was a statistically significant difference in quality of experience between network latencies, $\chi^2(4) = 17.280$, $p = 0.002$. The analysis of survey response data for player quality of experience produced a similar trend to the performance as network latency increased.

Our demographics analysis showed a few interesting trends among our participants that revealed which players were better at Neverball than others. Players who indicated that they had a lot of experience with video games performed better than players who did not. Players who indicated they had experience with hand-held and mobile games performed better than players who did not. There was however no significant difference between players who indicated they had experience with games like Neverball, and players who did not.

From this analysis, we conclude that playing thin client games with a network latency of less than 66 ms has little effect on the players' experience. The quality of players' experience decreases sharply between 66 ms and 100 ms, and remains low at latencies greater than 150 ms. Players who were more experienced with games performed better than less experienced players, but experience with games like Neverball did not significantly alter results.

Chapter 5: Conclusion

Thin client gaming services allow users who do not have access to powerful computers to play modern video games. Thin client gaming requires the user to have a sufficiently stable Internet connection to reliably stream the game. A common bottleneck in network connection quality is the roundtrip latency between client and server. Roundtrip latency increases the delay between when users send information and receive feedback. We conducted a study to find at what point latencies cause thin-client games to become difficult to play. We performed an experiment to map the relationship between latency, performance, and player Quality of Experience (QoE).

For our study, we set up a thin client game using a server computer and a client computer, Gaming Anywhere, and Neverball. Artificial latency was introduced in a controlled manner using Dummynet. We studied the effects of network latency on 34 participants. Each participant played five rounds of Neverball at different latencies and a practice round at the beginning with no artificial latency. We recorded participants' performance data as their time to complete the level. We recorded participants' subjective QoE data by asking them to answer five questions about each trial.

Participants' performed the best at latencies below 66ms, and worst at latencies above 150ms. Participants mean performance was statistically the same between 33ms and 66ms. Participants mean performance was also statistically the same above 150ms. There was a sharp decrease in average performance between 66ms and 150ms. Participants' QoE data also saw a sharp decrease between 66ms and 100ms, with a smaller but significant decrease between 100ms and 150ms.

In conclusion, playing thin client games with a network latency of less than 66ms has little effect on both the players' experience and performance. The quality of players' experience and performance decreases sharply between 66ms and 100ms, and remains low at latencies greater than 150ms. As such, developers and thin client gaming services should aim to keep the network latency between client and server below 66ms.

Chapter 6: Future Work

Our study was successful in finding a relationship between latency, player performance, and QoE. Nevertheless, various improvements to the study design may have improved the quality of the results. This section surmises the flaws in our design before moving on to discuss possible future work related to our study.

It is possible the subjects got bored after the first few trials and did not put as much effort into the later trials. While the repeated measures design allowed us to get a lot of data points out of individual subjects, gathering less data per subject could have given us higher quality data points. One possible solution for this would be to gather many more subjects, apply different latencies at random, and keep the number of trials each subject participates in to no more than three. This should minimize any boredom the players experience and any effect of learning between trials. With more subjects and smaller intervals between trial latencies, better results may be achieved. In addition, with more people, multivariate statistical analysis may yield trends between different subgroups. The majority of our participants were male, between ages 18 and 22, so analysis across gender and age was not fruitful. Taking a random sample from a more diverse population could result in more comprehensive data.

In the area of related work, it would be interesting to repeat a similar study with a different type of game. Other games may elicit different responses from the players possibly resulting in differences in the observed trends. In particular, repeating this study with a first person shooter could be revealing because the genre is both popular and commonly fast-paced. Another study with a slower-paced game may also be worthwhile.

Another area that could be improved is the survey response questions. Many of our questions elicited lackluster responses from the subjects. Subjects appear to have been more faithful in answering the first few questions than the last few. Additionally, some questions may have been too hard for the players to understand; the answers did not seem to correlate with the trends in latency.

References

Apperley, T. H. (2006). Genre and game studies: Toward a critical approach to video game genres. Simulation & Gaming*, 37*(1), 6-23. doi:10.1177/1046878105282278

Chen, K., Chang, Y., Hsu, H., Chen, D., Huang, C., & Hsu C. (2013). On the quality of service of cloud gaming systems.

Dueker, M., Noether, G. E., & SpringerLink ebooks - Mathematics and Statistics (Archive). (2012). Introduction to statistics: The nonparametric way. New York: Springer.

Everitt, B.,Mauchly test (2002). . Cambridge, United Kingdom: Cambridge University Press.

Huang, C., Chen, D., Hsu, C., & Chen, K. (2013). GamingAnywhere: An open-source cloud gaming testbed. Paper presented at the 827-830. doi:10.1145/2502081.2502222

Huang, C., Hsu, C., Chang, Y., and Chen, K. (2013). GamingAnywhere: An open cloud gaming system.

Jarschel, M., Schlosser, D., Scheuring, S., & Hossfeld, T. (2013). Gaming in the clouds: QoE and the users' perspective. Mathematical and Computer Modelling, 57(11-12), 2883-2894. doi:10.1016/j.mcm.2011.12.014

Lee, Y., Chen, K., Su, H., & Lei, C. (2012). Are all games equally cloud-gaming-friendly? An electromyographic approach.

Pantel, L., & Wolf, L. (2002). On the impact of delay on real-time multiplayer games. Paper presented at the 23-29. doi:10.1145/507670.507674

Pedersen, J. M., Riaz, M. T., Celestino, J., Dubalski, B., Ledzinski, D., & Patel, A. (2011). Assessing measurements of QoS for global cloud computing services. Paper presented at the 682-689. doi:10.1109/DASC.2011.120

Rutherford, A. (2012). ANOVA and ANCOVA: A GLM approach (2nd edition). Hoboken: Wiley-Blackwell.

Shea, R., Liu, J., Ngai, E. C. -., & Cui, Y. (2013). Cloud gaming: Architecture and performance. IEEE Network, 27(4), 16-21. doi:10.1109/MNET.2013.6574660

Urdan, T. C. (2010). Statistics in plain english Routledge.

Yu-Chun Chang, Po-Han Tseng, Kuan-Ta Chen, & Chin-Laung Lei. (2011). Understanding the performance of thin-client gaming. Paper presented at the 1-6. doi:10.1109/CQR.2011.5996092

Appendices

Appendix 1: Code

1.1: Dummynet Script

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
class SetLag
{
public:
     SetLag(void);
     ~SetLag(void);
     static void run(void);
};
SetLag::SetLag(void)
{
}


SetLag::~SetLag(void)
{
}

void SetLag::run(void)
{
     srand(time(NULL));
     printf("Run\n");
     int LAG=0;
     char COM1 [50];
     char COM2 [50];
     while (1)
     {
          printf("Enter the amount of lag[-1 = exit
program]\n");
          scanf("%d",&LAG);
          if (LAG == -1)
               break;
          sprintf(COM1,"ipfw pipe 1 config delay %dms\n",LAG/2);
          sprintf(COM2,"ipfw pipe 2 config delay %dms\n",LAG/2);
          printf("Setting lag to %dms\n",LAG);
          system(COM1);
          system("ipfw add 1337 pipe 1 ip from any to
192.168.255.002\n");
          system(COM2);
          system("ipfw add 1338 pipe 2 ip from 192.168.255.002
```

```
to any\n");
          system("pause\n");
          system("ipfw pipe 1 delete");
          system("ipfw delete 1337\n");
          system("ipfw pipe 2 delete\n");
          system("ipfw delete 1338\n");
     }
}

int main()
{
     SetLag::run();
     system("pause\n");
}
```

1.2: Gaming Anywhere Scripts

Client batch file:

```
cd \users\weird_000\Desktop\gaminganywhere-0.7.4\bin
ga-client config/client.abs.conf rtsp://192.168.1.2:8554/desktop
pause
```

Server batch file:

```
ga-server-event-driven config/server.neverball.conf
pause
```

1.3: Neverball configuration file

```
# configuration for the neverball game # work with ga-server-event driven
[core] include = common/server-common.conf include = common/controller.conf
include = common/video-x264.conf include = common/video-x264-param.conf
include = common/audio-lame.conf [video] video-fps = 50 [filter] filter-
source-pixelformat = rgba [ga-server-event-driven] game-exe =
D:\smallgame\NeverballPortable\App\Neverball\neverball.exe #game-resolution =
1280 720 # hook configuration # version: d9, d10, d10.1, d11, dxgi, sdl hook-
type = sdl enable-audio = true find-window-class = SDL_app enable-server-
rate-control = Y server-token-fill-interval = 20000 server-num-token-to-fill
= 1 server-max-tokens = 2
```

Retrieved from GamingAnywhere website at
http://gaminganywhere.org/dl/config/server.neverball.conf

Appendix 2: Procedure

2.1: Script

Thank you for volunteering for our study. We are going to have you play 6 rounds of a game called Neverball. The goal of this game is to reach the target at the end of the level as quickly as possible; please ignore the coins. Use the arrow keys to control the game. We will start by asking you to answer a few demographic questions. Then, at the end of every round, be asking you 5 short questions about the game. The questions will be the same for every round. If you have any questions about the game or the survey, please let us know.

2.2: Study Steps

1. As soon as the subject enters the room, have them sit at the computer.
2. Read the above script to him very clearly
3. Have the subject fill out the demographics section of the form.
4. Make sure the latency is set to 0ms, and have the subject play the level (practice)
5. Once finished, have the subject fill out the first part of the form
6. While the subject is filling out the form, set the latency to the desired amount
7. Have the subject play the level
8. Have the subject fill out the next section of the form and record the time it took for the subject to complete the level
9. Repeat steps 6-8 until all the subject's latencies have been played through

2.3: Frequently Asked Questions

Q: I found this awesome exploit!
A: Please try to keep the ball on the board at all times.

Q: Do you want me to collect the coins?
A: No. Please just focus on completing the level as fast as you can.

Q: What is the purpose of this experiment?
A: We are simply collecting data on how long it takes to complete a specific level in Neverball.

Q: I'm lost! Where is the goal?
A: Do your best to complete the level, unfortunately we cannot assist you.

Q: How many trials are there?
A: Six.

Q: What are the controls again?
A: Use the arrow keys on the keyboard to tilt the level and roll the ball to the goal.

Q: What does [word in survey] mean?
A: If you are unsure, just leave it blank

Q: Did you make the game?
A: No, we did not. It is a open-source game with many contributors.

Q: Can I change the resolution of the game?
A: Please leave the resolution settings where they are.

Q: Can I change the volume of the game?
A: Unfortunately there are no speakers hooked up to this computer, so we cannot turn the sound on.

Q: I have a question about this setup and/or cloud gaming.
A: We will answer all setup and network-related questions at the end of the study.

2.4: Server Computer Specs

Intel core i7 2.4 GHz CPU
Nvidia GeForce GTX 675M GPU
16 GB RAM
750 GB Hard Drive

Appendix 3: Data

	Time							Question 1 - QoE					
	0	33	66	100	150	200	Average	0 ms	33 ms	66 ms	100 ms	150 ms	200 ms
1	31.75	17.29	13.44	29.42	31.17	26.58	24.94167	4	5	5	4	4	4
2	43.77	14.91	15.59	33.91	19.94	20.42	24.75667	3	5	5	2	3	3
3	41.96	32.27	33.27	27.09	16.49	55.11	34.365	3	3	3	4	3	3
4		13.22	29.22	48.8	41.02	33.95	33.242	3	5	4	4	4	4
5	24.45	60	17.22	34.7	58.11	54.72	41.53333	3	1	2	2	1	1
6	43.44	14.7	18.2	40.79	31.92	51.17	33.37	2	3	5	3	2	3
7	36.69	20.64	33.47	58.77	36.39	48.52	39.08	4	2	4	4	4	3
8	7.69	21.67	20.23	16.61	43.29	23.29	22.13	2	3	4	5	4	5
9	47.86	73.26	70.58		90	51.62	66.664	3	4	3	3	2	4
10	26.41	28.22	23.88	27.12	46.25	49.23	33.51833	1	2	2	1	2	1
11	27.4	31.81	16.95	35.82	18.16	42.47	28.76833	3	3	3	3	3	2
12	26.9	13.34	20.07	24.89	40.86	45.49	28.59167	3	3	4	5	3	4
13	45.89	27.35	15.75	30.57	17.4	41.24	29.7	3	4	4	3	3	4
14		16.09	39.23	37.84	26.74	25.97	29.174	4	4	2	4	4	4
15		20.93	25.41	22.49	23.25	23.43	23.102	2	4	3	4	3	3
16	35.99	12.35	14.1	27.35	30.27	21.79	23.64167	3	5	4	4	4	4
17	65.17	38.09	30.2	27.35	36.43	46.7	40.65667	3	4	3	4	2	3
18	23.42	18.73	10.78	15.19	29.43	30.17	21.28667	3	4	4	4	3	2
19		29	77.35		67.37	45.98	54.925	3	3	3	4	3	3
20		25.77	25.5	36.66	57.52	34.99	36.088	2	3	3	3	3	3
21		23.68	32.78	58.24	51.5	90	51.24	2	4	3	4	2	1
22	27.41	21.4	23.4	20.46	37.43	22.27	25.395	4	4	4	3	4	4
23	19.66	19.47	13.96	23.5	37.21	24.01	22.96833	4	3	4	3	4	4
24	18.26	35.18	20.73	30.8	19.48	25.64	25.015	2	2	2	2	2	2
25	29.54	25.98	26.27	14.07	62.57	26.2	30.77167	3	3	3	3	2	3
26	49.59	90	90	89.75	90	90	83.22333	4	3	4	4	3	3
27	14.36	18.8		49.73		40.11	30.75	3	2	2	3	3	3
28	21.3		41.67		11.09	32.33	26.5975	4	4	3	3	5	4
29	30.84	17.76	20.51	30.82	35.41	21.23	26.095	2	5	5	5	3	5
30	41.32	23.31	19.83	25.75	55.06	60.75	37.67	5	5	5	4	4	4
31	21.72	37.42	12.76	48.29	36.61	90	41.13333	3	4	4	3	4	2
32	29.4	18.84	24.22	22.27	64.44	41.41	33.43	5	5	5	5	5	5
33	73.33	17.42	76.68	57.68	45.56	17.21	47.98	3	3	2	3	2	2
34	77.3	14.5	28.42	72.56	49.22	32.53	45.755	1	2	2	2	1	2
Average	35.10071	27.0727	29.7476	36.1061	41.1391	40.7803	35.22233	3	3.5	3.47059	3.44118	3.05882	3.14706

Average	Question 2 - Performance						Average	Question 3 - Responsiveness					
	0 ms	33 ms	66 ms	100 ms	150 ms	200 ms		0 ms	33 ms	66 ms	100 ms	150 ms	200 ms
4.33333	2	4	4	3	3	4	3.33333	2	3	4	4	4	4
3.5	2	5	5	3	4	4	3.83333	5	5	5	1	2	2
3.16667	2	2	3	3	3	2	2.5	3	3	3	3	3	3
4	1	5	4	2	2	3	2.83333	2	4	4	3	3	4
1.66667	3	1	2	2	1	1	1.66667	2	1	1	1	1	1
3	3	4	3	4	3	3	3.33333	4	4	4	4	3	3
3.5	2	4	3	2	3	3	2.83333	4	4	4	4	4	4
3.83333	5	1	3	4	3	4	3.33333	1	2	3	4	4	4
3.16667	2	4	3	2	2	4	2.83333	3	3	2	4	1	4
1.5	2	3	4	1	2	1	2.16667	2	2	2	2	1	1
2.83333	2	3	4	3	4	3	3.16667	4	3	4	3	3	3
3.66667	3	3	4	5	2	4	3.5	4	3	4	5	3	4
3.5	2	3	4	2	3	3	2.83333	3	3	3	3	3	3
3.66667	1	5	2	3	3	3	2.83333	2	3	2	4	3	4
3.16667	2	3	3	4	2	2	2.66667	5	4	3	4	2	2
4	2	5	5	3	3	4	3.66667	5	5	5	4	3	4
3.16667	2	4	3	4	4	3	3.33333	4	4	2	4	3	3
3.33333	3	4	4	4	4	3	3.66667	2	3	4	4	2	2
3.16667	1	3	2	5	3	3	2.83333	3	4	4	3	3	3
2.83333	1	4	4	3	3	3	3	5	3	3	4	4	4
2.66667	1	4	4	3	2	2	2.66667	1	3	3	2	2	2
3.83333	3	4	4	2	3	4	3.33333	5	5	5	5	5	5
3.66667	5	5	5	5	4	5	4.83333	3	3	3	3	3	3
2	1	3	4	3	4	3	3	5	5	5	5	5	5
2.83333	2	3	4	4	1	3	2.83333	2	2	2	3	2	2
3.5	3	2	3	3	2	2	2.5	4	2	3	4	2	2
2.66667	1	2	1	1	1	3	1.5	2	1	1	2	3	3
3.83333	2	3	1	2	5	3	2.66667	4	4	2	4	5	3
4.16667	3	5	5	4	4	5	4.33333	4	5	5	5	4	5
4.5	2	4	5	5	2	2	3.33333	5	5	4	5	4	3
3.33333	2	3	4	2	3	1	2.5	3	4	4	3	4	3
5	3	5	4	4	3	4	3.83333	5	5	5	3	5	3
2.5	1	3	3	1	1	4	2.16667	2	4	4	3	4	4
1.66667	1	3	2	1	2	2	1.83333	2	3	2	1	1	1
3.26961	2.14706	3.5	3.47059	3	2.76471	3.02941	2.98529	3.29412	3.44118	3.35294	3.41176	3.05882	3.11765

Question 4 - Responsiveness to Performance							Question 5 - Enjoyable						
Average	0 ms	33 ms	66 ms	100 ms	150 ms	200 ms	Average	0 ms	33 ms	66 ms	100 ms	150 ms	200 ms
3.5	3	4	4	5	5	5	4.33333	5	5	5	5	5	4
3.33333	1	1	1	5	4	4	2.66667	2	3	3	2	2	2
3	4	5	5	5	5	5	4.83333	2	3	3	3	3	3
3.33333	4	4	3	3	3	3	3.33333	3	4	4	4	4	3
1.16667	4	5	4	4	5	5	4.5	2	1	2	2	1	1
3.66667	1	1	1	1	2	1	1.16667	4	3	3	3	3	3
4	4	4	4	4	4	4	4	4	4	4	4	4	4
3	5	5	4	1	4	2	3.5	2	3	3	4	3	4
2.83333	3	3	4	2	5	2	3.16667	3	3	3	4	1	4
1.66667	2	4	5	4	4	5	4	1	1	2	1	1	1
3.33333	3	2	3	3	3	3	2.83333	2	2	2	2	3	2
3.83333	2	3	2	2	4	1	2.33333	3	3	4	4	2	3
3	4	4	4	4	4	4	4	4	4	4	4	4	4
3	5	3	4	5	3	4	4	5	4	3	4	4	3
3.33333	1	1	2	1	4	4	2.16667	3	4	3	4	3	4
4.33333	4	4	4	5	4	4	4.16667	3	5	4	4	4	3
3.33333	2	3	3	4	2	2	2.66667	3	4	3	4	3	4
2.83333	4	3	3	5	3	3	3.5	3	3	4	4	3	3
3.33333	1	3	4	3	2	3	2.66667	3	3	3	4	3	3
3.83333	4	4	3	3	3	3	3.33333	2	3	3	3	3	4
2.16667	5	4	4	4	4	5	4.33333	1	3	3	2	2	1
5	4	3	4	4	4	3	3.66667	4	3	3	3	3	3
3	3	3	2	2	4	3	2.83333	4	3	3	3	4	3
5	5	5	5	5	5	5	5	2	1	1	1	2	2
2.16667	4	4	4	4	4	4	4	2	3	3	3	2	3
2.83333	4	5	4	5	4	4	4.33333	5	3	4	4	3	3
2	5	5	5	5	4	4	4.66667	2	3	2	2	3	3
3.66667	4	4	5	4	5	5	4.5	3	4	3	3	4	4
4.66667	4	4	4	4	4	4	4	3	4	4	3	4	4
4.33333	3	3	2	4	3	4	3.16667	4	4	4	4	3	3
3.5	4	4	3	5	4	5	4.16667	3	4	4	3	3	2
4.33333	4	5	5	4	4	4	4.33333	4	4	4	4	4	4
3.5	2	2	2	2	2	2	2	2	2	2	2	2	2
1.66667	5	4	5	5	5	5	4.83333	2	1	1	2	1	1
3.27941	3.44118	3.55882	3.55882	3.70588	3.79412	3.64706	3.61765	2.94118	3.14706	3.11765	3.17647	2.91176	2.94118

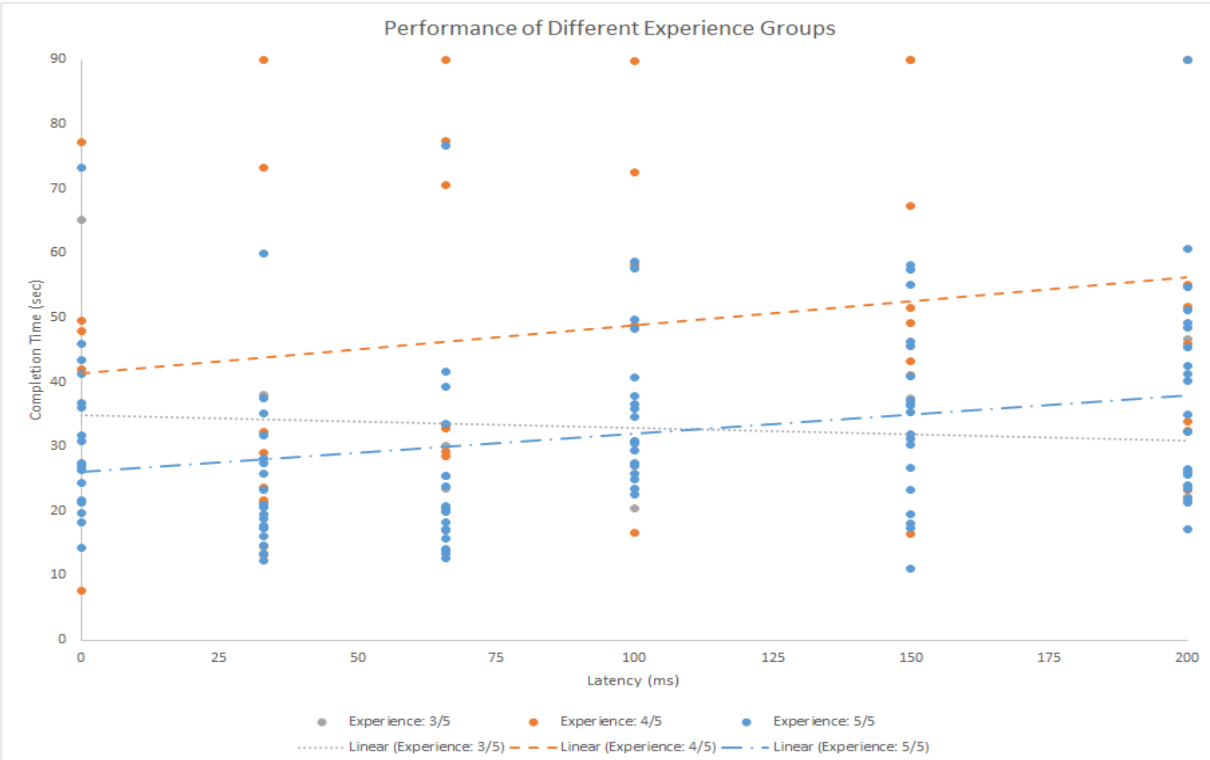
Demographics

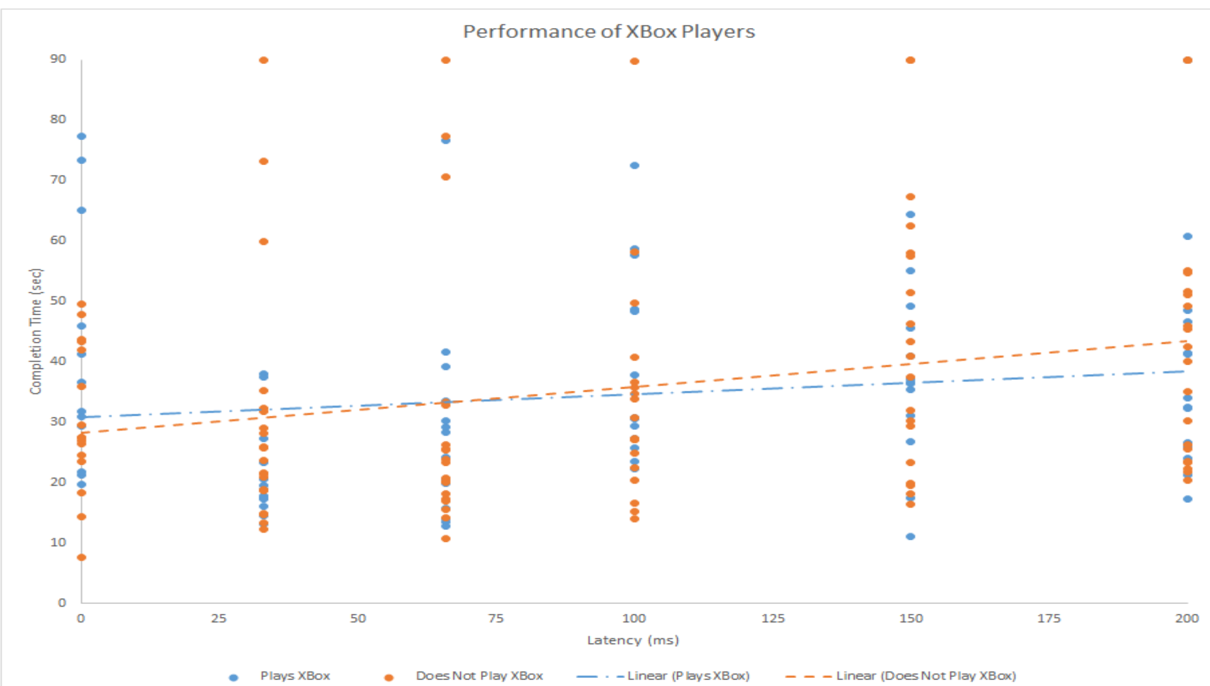
Average	Experience	Xbox	Playstation	Nintendo	Handheld	Computer	Mobile	Action	Adventure	Arcade	Roleplaying	Simulation
4.83333	5	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
2.33333		FALSE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
2.83333	4	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
3.66667	4	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE
1.5	5	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE
3.16667	5	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
4	5	TRUE	FALSE	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE
3.16667	4	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
3	4	FALSE	FALSE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE
1.16667	5	FALSE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE
2.16667	5	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
3.16667	5	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE
4	5	TRUE	TRUE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE
3.83333	5	TRUE	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE
3.5	5	FALSE	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	FALSE	TRUE	FALSE	FALSE
3.83333	5	FALSE	FALSE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE
3.5	3	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	FALSE
3.33333		FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE
3.16667	4	FALSE	TRUE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE
3	5	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	FALSE	TRUE
2	4	FALSE	FALSE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
3.16667	3	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE	FALSE
3.33333	5	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
1.5	5	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE
2.66667		FALSE	TRUE	FALSE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	FALSE
3.66667	4	FALSE	FALSE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	FALSE	FALSE	FALSE
2.5	5	FALSE	FALSE	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE
3.5	5	TRUE	FALSE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE	FALSE	TRUE	TRUE
3.66667	5	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
3.66667	5	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE
3.16667	5	TRUE	TRUE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE	TRUE
4		TRUE	FALSE	TRUE	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE
2	5	TRUE	FALSE	FALSE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE
1.33333	4	TRUE	TRUE	TRUE	FALSE	TRUE	FALSE	TRUE	TRUE	TRUE	TRUE	TRUE
3.03922	4.6											

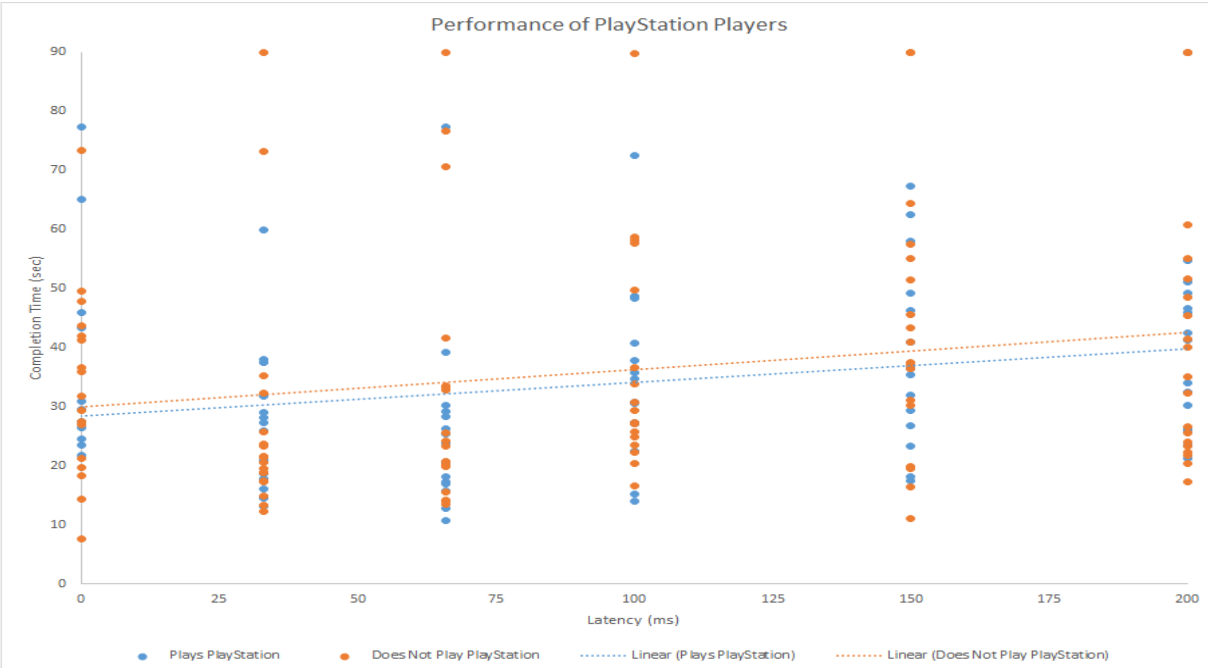
Strategy	Sports	Other	Age	Gender	Played Before	Why Participate
TRUE	TRUE		18	M	TRUE	Email, Gift card, Cloud Gaming
TRUE	TRUE		21	M	TRUE	Email
TRUE	TRUE		19	M	TRUE	Email
TRUE	FALSE		19	M	FALSE	Email, Friend
TRUE	FALSE		20	M	TRUE	Email, Cloud Gaming
TRUE	TRUE		19	M	TRUE	Email, Gift card, Cloud Gaming
TRUE	TRUE		21	M	TRUE	Friend
TRUE	TRUE	Rogue-Like	21	M	TRUE	Empathy
TRUE	FALSE		19	M	TRUE	Friend
TRUE	TRUE		20	F	TRUE	Gift Card
TRUE	TRUE		18	M	TRUE	Email, Gift card, Cloud Gaming
TRUE	TRUE		19	M	TRUE	Email, Cloud Gaming
TRUE	FALSE		19	M	TRUE	Email
TRUE	TRUE		21	M	FALSE	Friend
FALSE	FALSE	Shooter	19	M	TRUE	Gift Card
FALSE	FALSE		20	M	TRUE	Email
TRUE	TRUE		20	M	FALSE	Email
TRUE	FALSE		19	M	TRUE	Email, Cloud Gaming
TRUE	TRUE	Puzzle	20	M	FALSE	Friend
TRUE	FALSE		18	M	TRUE	Email, Cloud Gaming
TRUE	TRUE		19	M	TRUE	Email
TRUE	FALSE		20	M	TRUE	Friend
TRUE	FALSE		19	M	TRUE	Friend
TRUE	FALSE		22	M	FALSE	Email, Cloud Gaming
TRUE	FALSE		19	M	FALSE	Friend
TRUE	FALSE		20	F	TRUE	Friend, Empathy
TRUE	FALSE		21	M	TRUE	Friend
TRUE	FALSE		20	M	FALSE	Friend
TRUE	FALSE		19	M	TRUE	Email
TRUE	TRUE		20	M	FALSE	Email
TRUE	TRUE		19	M	TRUE	Email, Cloud Gaming, Empathy
TRUE	FALSE		19	M	TRUE	Email
FALSE	FALSE		20	M	TRUE	Friend
TRUE	FALSE		20	F	TRUE	Email
			19.6176			

Appendix 4: Graphs and Analysis

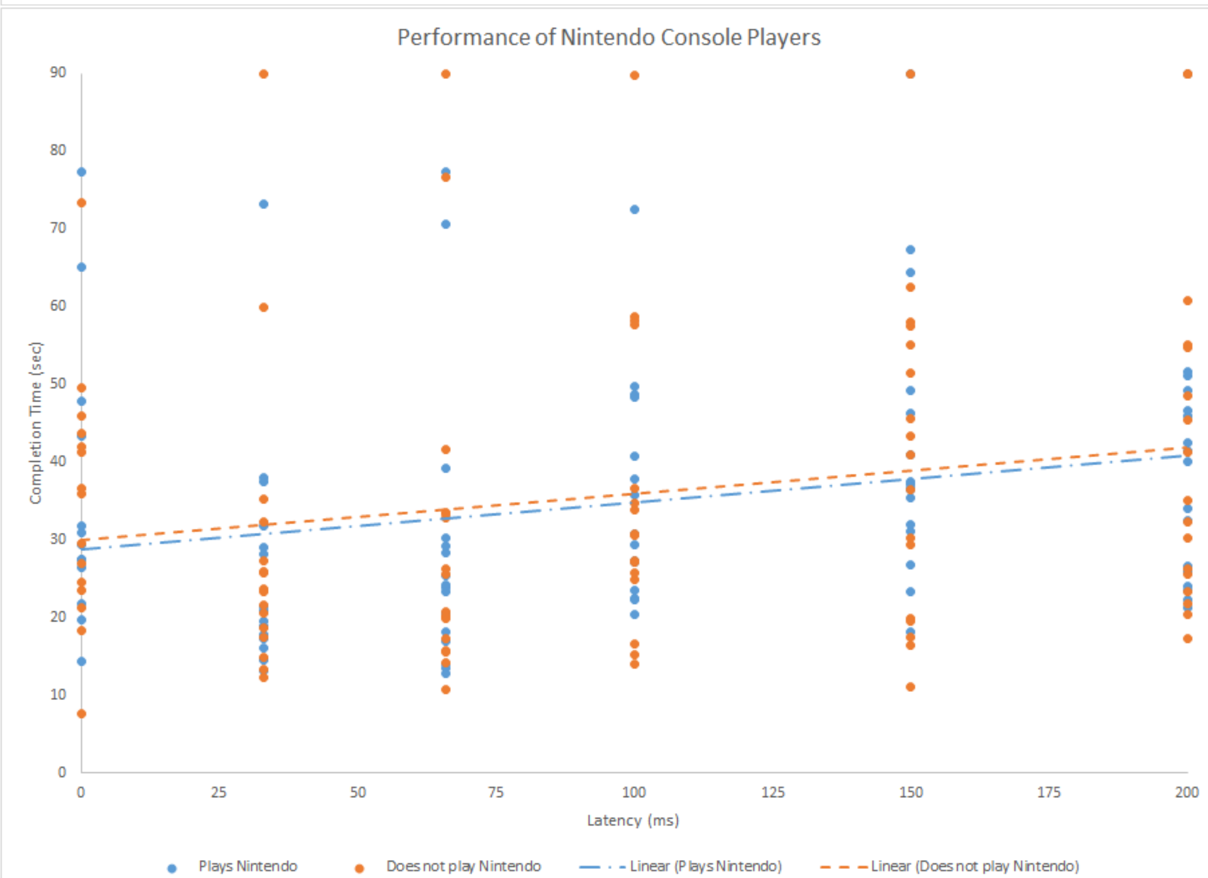
4.1: Demographics



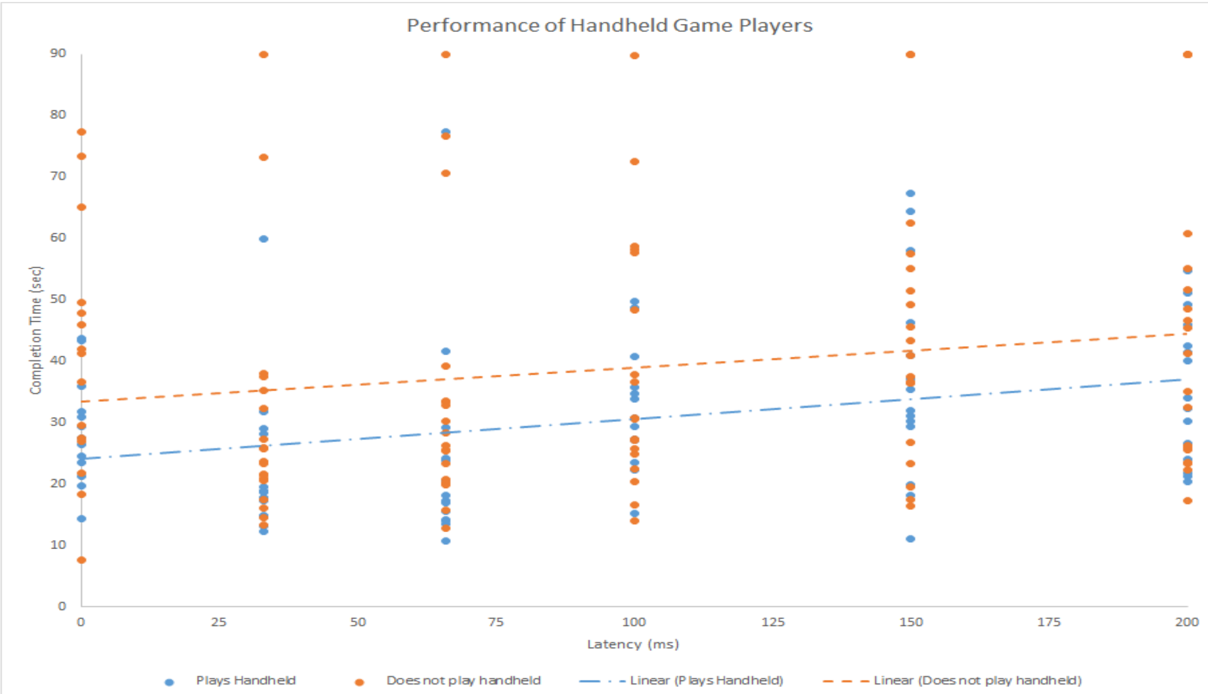


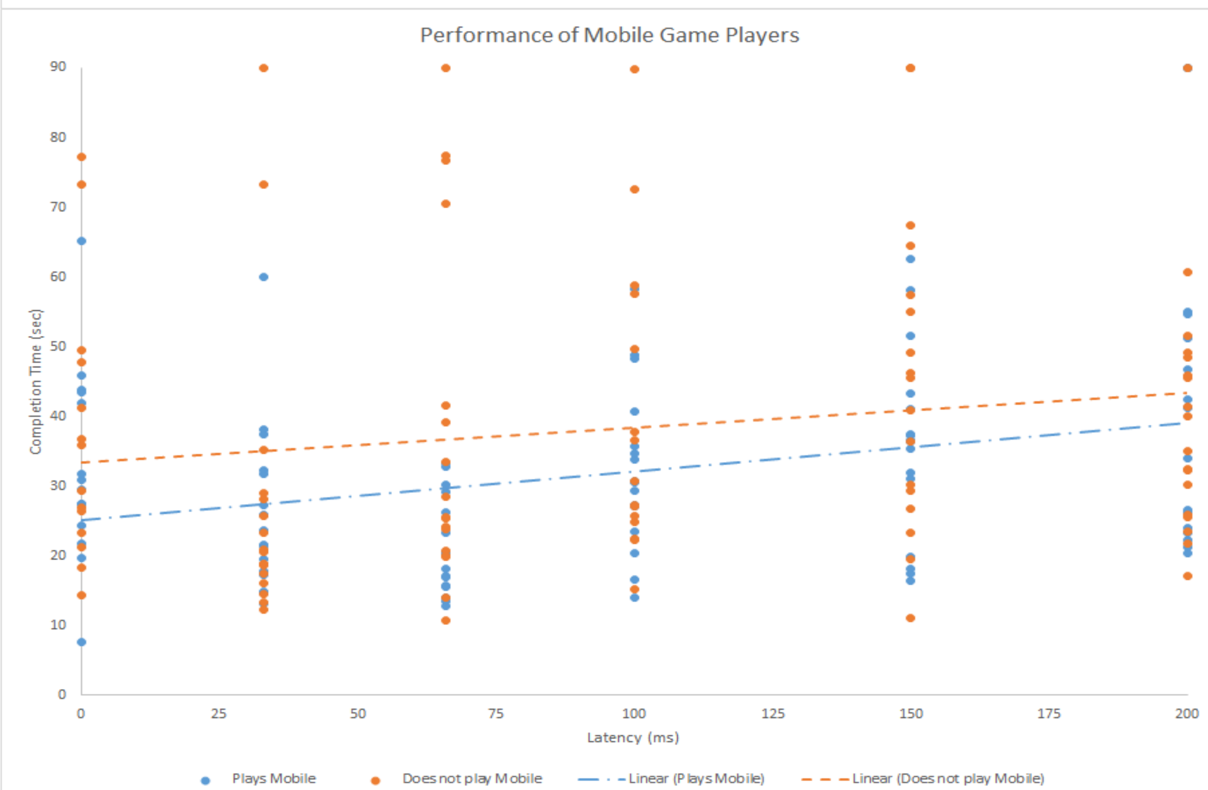


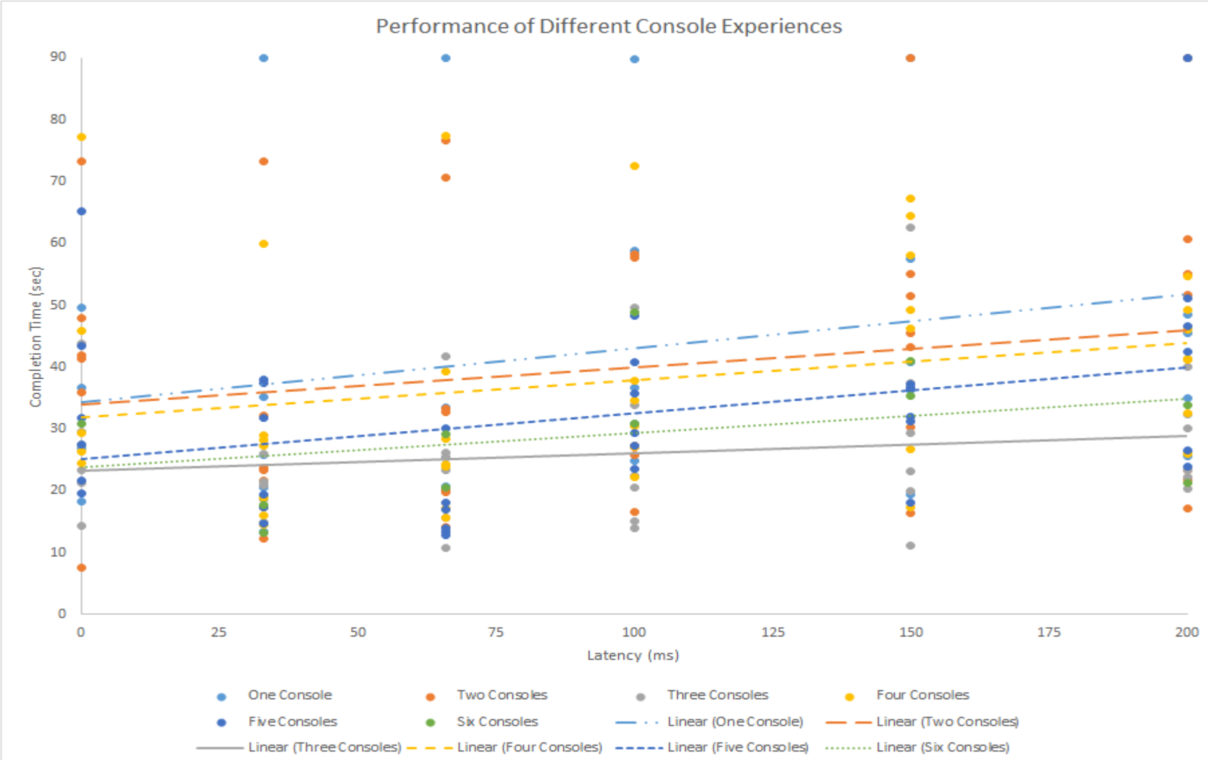
Performance of PlayStation Players
Completion Time (sec)
0
10
20
30
40
50
60
70
80
90
0
25
50
75
100
125
150
175
200
Latency (ms)
Plays PlayStation
Does Not Play PlayStation
Linear (Plays PlayStation)
Linear (Does Not Play PlayStation)

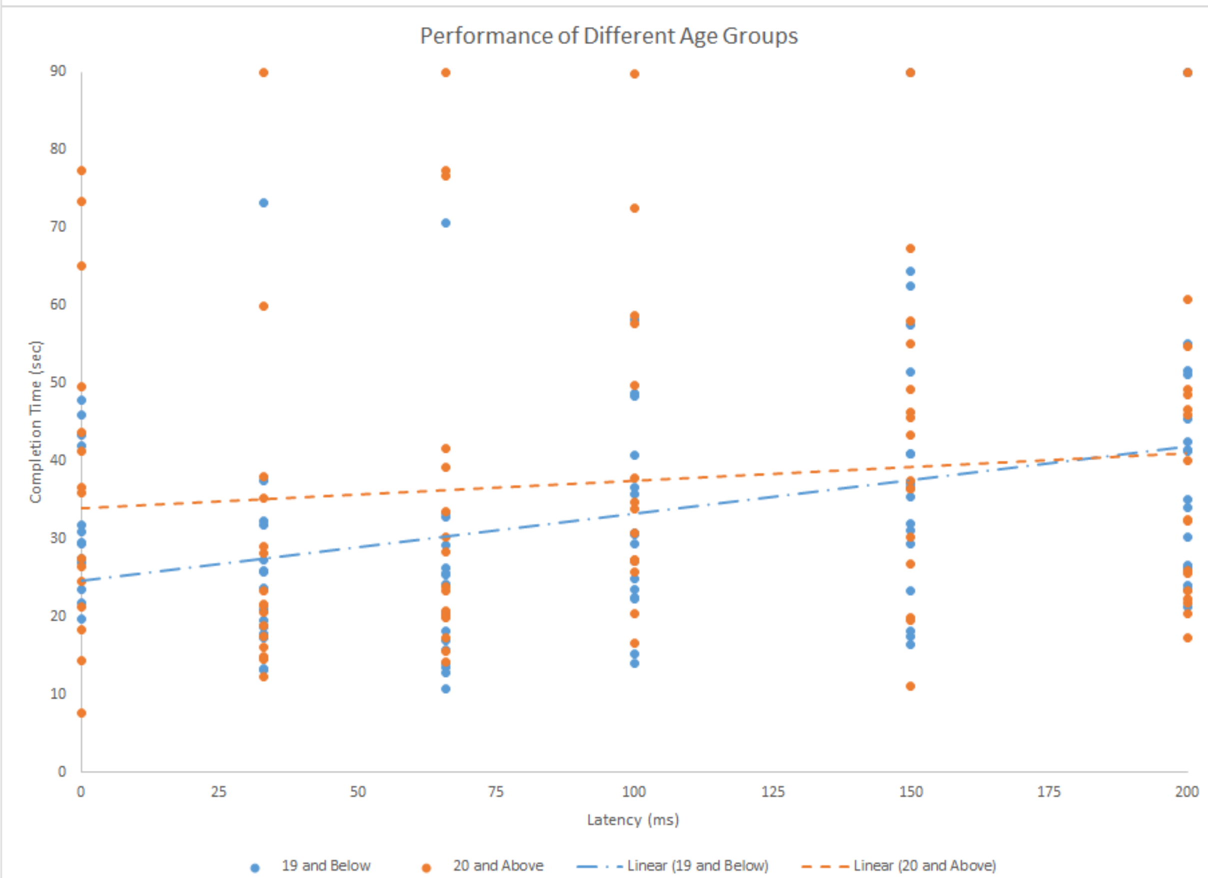


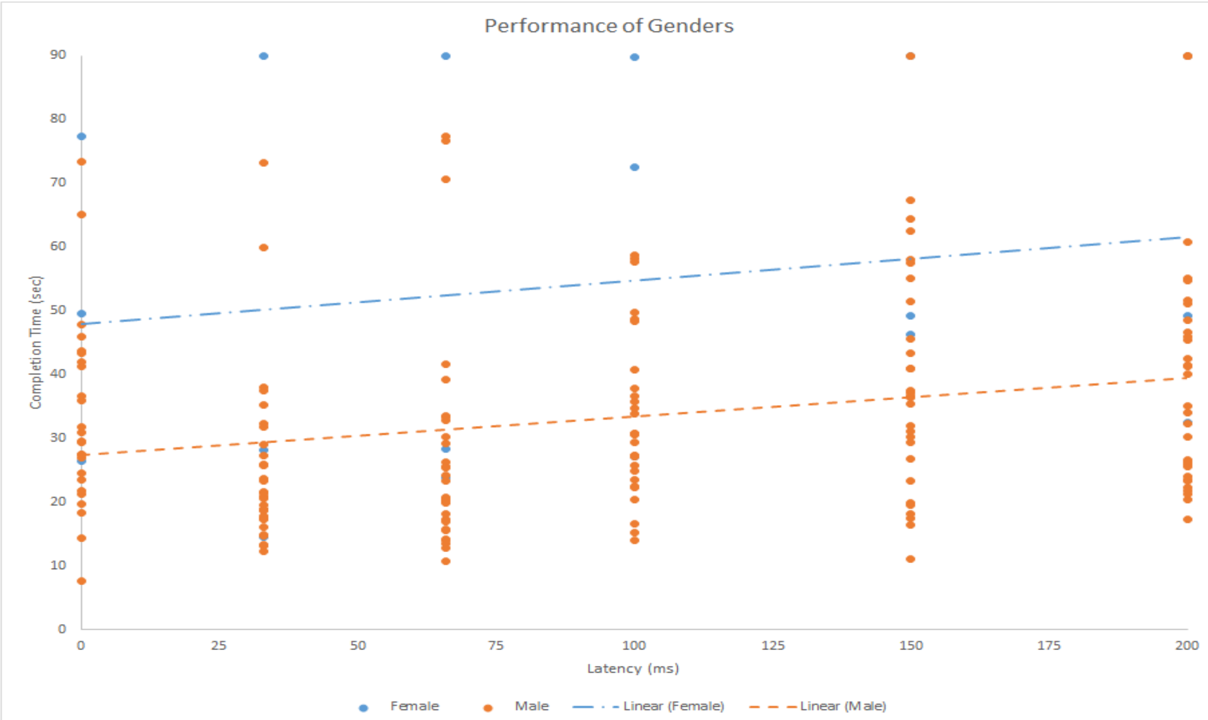
Performance of Nintendo Console Players
Completion Time (sec)
0
10
20
30
40
50
60
70
80
90
0
25
50
75
100
125
150
175
200
Latency (ms)
Plays Nintendo
Does not play Nintendo
Linear (Plays Nintendo)
Linear (Does not play Nintendo)

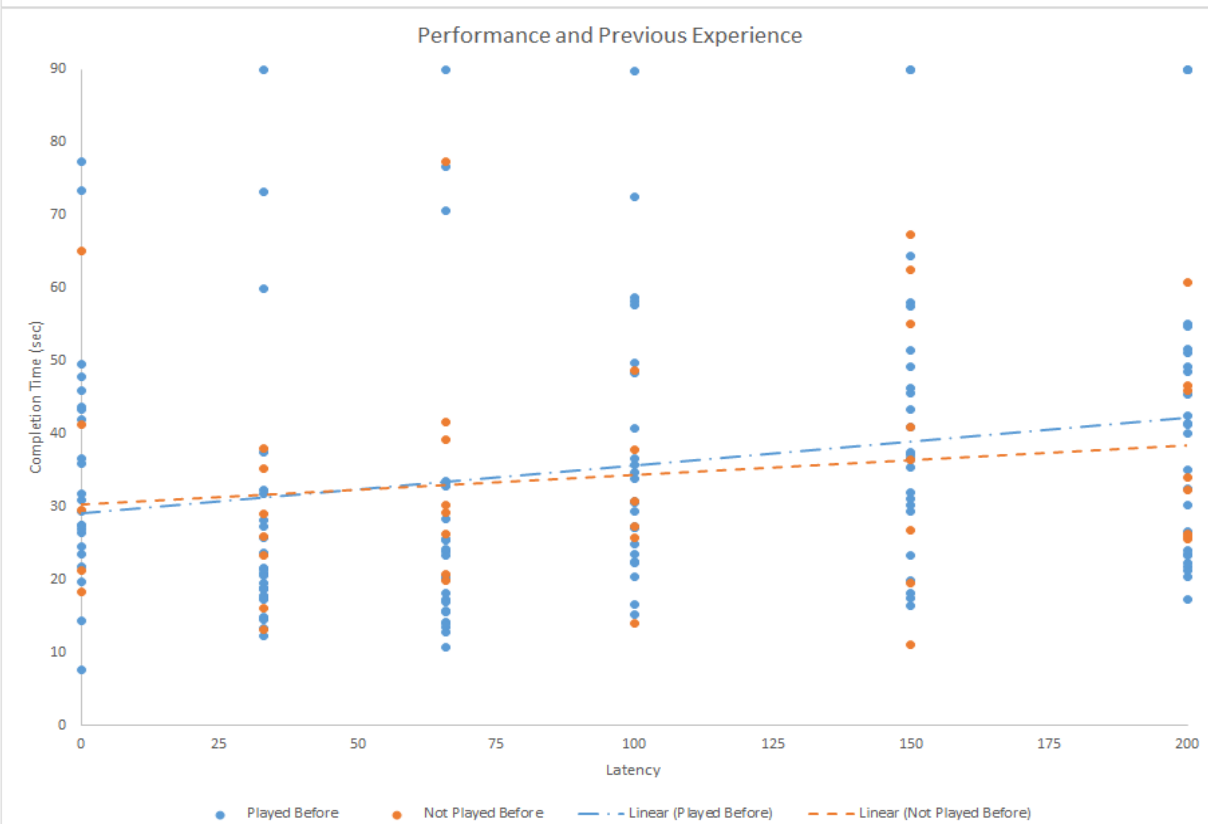

Performance of Handheld Game Players
Completion Time (sec)
Latency (ms)
Plays Handheld
Does not play handheld
Linear (Plays Handheld)
Linear (Does not play handheld)

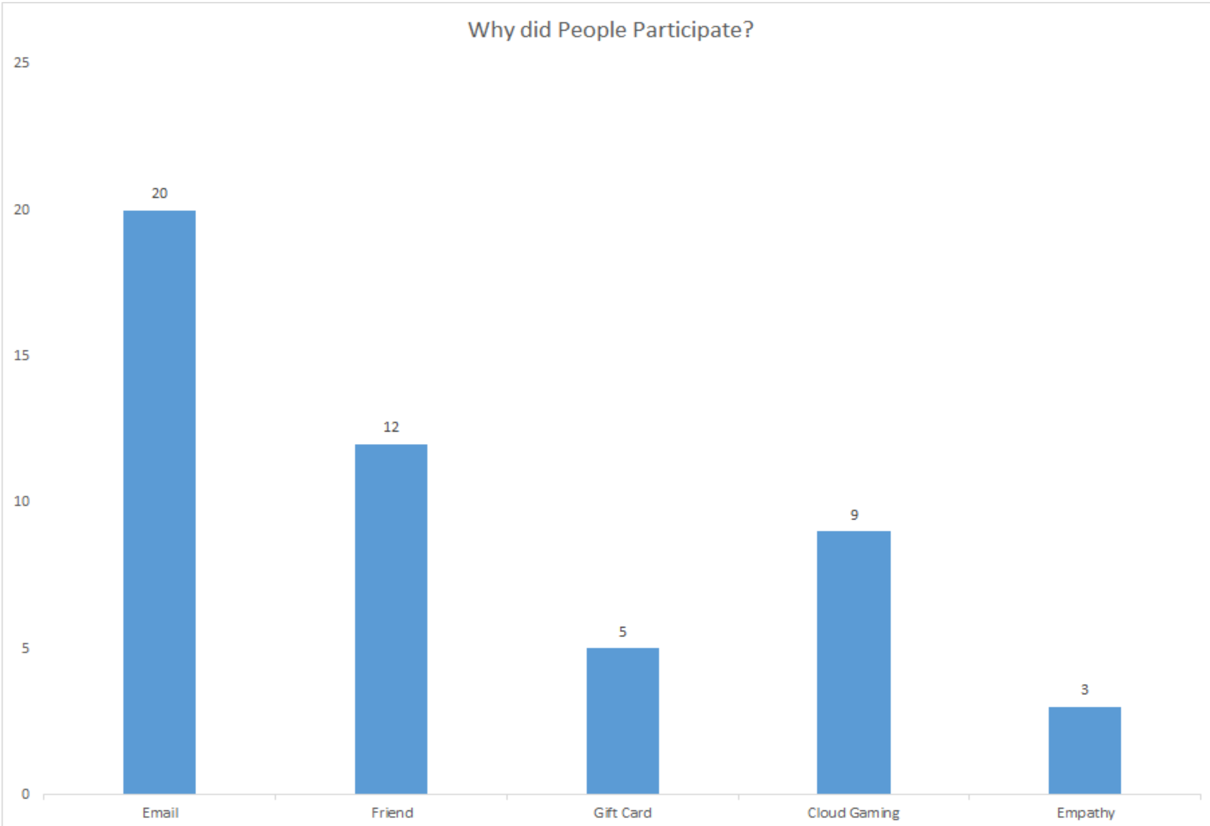

Performance of Mobile Game Players
Completion Time (sec)
Latency (ms)
Plays Mobile
Does not play Mobile
Linear (Plays Mobile)
Linear (Does not play Mobile)


Performance of Different Console Experiences
Completion Time (sec)
Latency (ms)
0
10
20
30
40
50
60
70
80
90
0
25
50
75
100
125
150
175
200
One Console
Two Consoles
Three Consoles
Four Consoles
Five Consoles
Six Consoles
Linear (One Console)
Linear (Two Consoles)
Linear (Three Consoles)
Linear (Four Consoles)
Linear (Five Consoles)
Linear (Six Consoles)


Performance of Different Age Groups
Completion Time (sec)
Latency (ms)
0
10
20
30
40
50
60
70
80
90
0
25
50
75
100
125
150
175
200
19 and Below
20 and Above
Linear (19 and Below)
Linear (20 and Above)


Performance of Genders
Completion Time (sec)
90
80
70
60
50
40
30
20
10
0
0
25
50
75
100
125
150
175
200
Latency (ms)
Female
Male
Linear (Female)
Linear (Male)


Performance and Previous Experience
Completion Time (sec)
90
80
70
60
50
40
30
20
10
0
0
25
50
75
100
125
150
175
200
Latency
Played Before
Not Played Before
Linear (Played Before)
Linear (Not Played Before)



Why did People Participate?
25
20
15
10
5
0
20
12
5
9
3
Email
Friend
Gift Card
Cloud Gaming
Empathy

4.2: Performance and QoE (SPSS Output)

```
EXAMINE VARIABLES=LATENCY_00 LATENCY_33 LATENCY_66 LATENCY_100
LATENCY_150 LATENCY_200
  /PLOT BOXPLOT HISTOGRAM NPPLOT
  /COMPARE GROUPS
  /STATISTICS DESCRIPTIVES
  /CINTERVAL 95
  /MISSING LISTWISE
  /NOTOTAL.
```

NPar Tests

Notes

Output Created		04-MAY-2014 13:04:14
Comments		
Input	Data	\\filer\home\My_Documents\encrypt7500632813994431560.spv.sav
	Active Dataset	DataSet1
	Filter	<none>
	Weight	<none>
	Split File	<none>
	N of Rows in Working Data File	34
Missing Value Handling	Definition of Missing	User-defined missing values are treated as missing.
	Cases Used	Statistics for all tests are based on cases with no missing data for any variables used.
Syntax		NPAR TESTS /FRIEDMAN=QOE_33 QOE_66 QOE_100 QOE_150 QOE_200 /STATISTICS DESCRIPTIVES /MISSING LISTWISE.
Resources	Processor Time	00:00:00.02
	Elapsed Time	00:00:00.11
	Number of Cases Allowed[a]	78643

a. Based on availability of workspace memory.

Descriptive Statistics

	N	Mean	Std. Deviation	Minimum	Maximum
QOE_33	34	3.5000	1.08012	1.00	5.00
QOE_66	34	3.4706	1.02204	2.00	5.00
QOE_100	34	3.4412	.95952	1.00	5.00

QOE_150	34	3.0588	1.01328	1.00	5.00
QOE_200	34	3.0588	1.01328	1.00	5.00

Friedman Test

Ranks

	Mean Rank
QOE_33	3.40
QOE_66	3.31
QOE_100	3.26
QOE_150	2.51
QOE_200	2.51

Test Statistics[a]

N	34
Chi-Square	17.280
df	4
Asymp. Sig.	.002

a. Friedman Test

```
NPAR TESTS
  /FRIEDMAN=PERF_33 PERF_66 PERF_100 PERF_150 PERF_200
  /STATISTICS DESCRIPTIVES QUARTILES
  /MISSING LISTWISE.
```

NPar Tests

Notes

Output Created		19-APR-2014 21:48:07
Comments		
Input	Data	\\filer\home\My_Documents\encrypt7500632813994431560.spv.sav
	Active Dataset	DataSet1
	Filter	<none>
	Weight	<none>
	Split File	<none>
	N of Rows in Working Data File	34
Missing Value Handling	Definition of Missing	User-defined missing values are treated as missing.

	Cases Used	Statistics for all tests are based on cases with no missing data for any variables used.
Syntax		NPAR TESTS /FRIEDMAN=PERF_33 PERF_66 PERF_100 PERF_150 PERF_200 /STATISTICS DESCRIPTIVES QUARTILES /MISSING LISTWISE.
Resources	Processor Time	00:00:00.02
	Elapsed Time	00:00:00.01
	Number of Cases Allowed[a]	78643

a. Based on availability of workspace memory.

Descriptive Statistics

	N	Mean	Std. Deviation	Minimum	Maximum	Percentiles	
						25th	50th (Median)
PERF_33	34	3.5000	1.10782	1.00	5.00	3.0000	3.5000
PERF_66	34	3.4706	1.07971	1.00	5.00	3.0000	4.0000
PERF_100	34	3.0000	1.20605	1.00	5.00	2.0000	3.0000
PERF_150	34	2.7647	1.01679	1.00	5.00	2.0000	3.0000
PERF_200	34	3.0294	1.02942	1.00	5.00	2.0000	3.0000

Descriptive Statistics

	Percentiles
	75th
PERF_33	4.0000
PERF_66	4.0000
PERF_100	4.0000
PERF_150	3.2500
PERF_200	4.0000

Friedman Test

Ranks

	Mean Rank
PERF_33	3.56
PERF_66	3.54
PERF_100	2.75
PERF_150	2.40
PERF_200	2.75

Test Statistics[a]

N	34
Chi-Square	18.277
df	4
Asymp. Sig.	.001

a. Friedman Test

```
NPAR TESTS
  /FRIEDMAN=Responsiveness_33 Responsiveness_66 Responsiveness_100
Responsiveness_150 Responsiveness_200
  /STATISTICS DESCRIPTIVES QUARTILES
  /MISSING LISTWISE.
```

NPar Tests

Notes

Output Created		19-APR-2014 21:52:48
Comments		
Input	Data	\\filer\home\My_Documents\encrypt7500632813994431560.spv.sav
	Active Dataset	DataSet1
	Filter	<none>
	Weight	<none>
	Split File	<none>
	N of Rows in Working Data File	34
Missing Value Handling	Definition of Missing	User-defined missing values are treated as missing.
	Cases Used	Statistics for all tests are based on cases with no missing data for any variables used.
Syntax		NPAR TESTS /FRIEDMAN=Responsiveness_33 Responsiveness_66 Responsiveness_100 Responsiveness_150 Responsiveness_200 /STATISTICS DESCRIPTIVES QUARTILES /MISSING LISTWISE.
Resources	Processor Time	00:00:00.02
	Elapsed Time	00:00:00.01
	Number of Cases Allowed[a]	78643

a. Based on availability of workspace memory.

Descriptive Statistics

	N	Mean	Std. Deviation	Minimum	Maximum	Percentiles 25th
Responsiveness_33	34	3.4412	1.13328	1.00	5.00	3.0000
Responsiveness_66	34	3.3529	1.17763	1.00	5.00	2.0000
Responsiveness_100	34	3.4118	1.13131	1.00	5.00	3.0000
Responsiveness_150	34	3.0588	1.17914	1.00	5.00	2.0000

Responsiveness_200	34	3.1176	1.09447	1.00	5.00	2.0000

Descriptive Statistics

	Percentiles	
	50th (Median)	75th
Responsiveness_33	3.0000	4.0000
Responsiveness_66	3.5000	4.0000
Responsiveness_100	4.0000	4.0000
Responsiveness_150	3.0000	4.0000
Responsiveness_200	3.0000	4.0000

Friedman Test

Ranks

	Mean Rank
Responsiveness_33	3.24
Responsiveness_66	3.19
Responsiveness_100	3.24
Responsiveness_150	2.60
Responsiveness_200	2.74

Test Statistics[a]

N	34
Chi-Square	8.297
df	4
Asymp. Sig.	.081

a. Friedman Test

```
NPAR TESTS
  /FRIEDMAN=RespToPerf_33 RespToPerf_66 RespToPerf_100 RespToPerf_150
RespToPerf_200
  /STATISTICS DESCRIPTIVES QUARTILES
  /MISSING LISTWISE.
```

NPar Tests

Notes

Output Created	19-APR-2014 21:53:27
Comments	

Input	Data	\\filer\home\My_Documents\encrypt750 0632813994431560.spv.sav
	Active Dataset	DataSet1
	Filter	<none>
	Weight	<none>
	Split File	<none>
	N of Rows in Working Data File	34
Missing Value Handling	Definition of Missing	User-defined missing values are treated as missing.
	Cases Used	Statistics for all tests are based on cases with no missing data for any variables used.
Syntax		NPAR TESTS /FRIEDMAN=RespToPerf_33 RespToPerf_66 RespToPerf_100 RespToPerf_150 RespToPerf_200 /STATISTICS DESCRIPTIVES QUARTILES /MISSING LISTWISE.
Resources	Processor Time	00:00:00.02
	Elapsed Time	00:00:00.01
	Number of Cases Allowed[a]	78643

a. Based on availability of workspace memory.

Descriptive Statistics

	N	Mean	Std. Deviation	Minimum	Maximum	Percentiles 25th
RespToPerf_33	34	3.5588	1.15971	1.00	5.00	3.0000
RespToPerf_66	34	3.5588	1.15971	1.00	5.00	3.0000
RespToPerf_100	34	3.7059	1.29168	1.00	5.00	3.0000
RespToPerf_150	34	3.7941	.91385	2.00	5.00	3.0000
RespToPerf_200	34	3.6471	1.17763	1.00	5.00	3.0000

Descriptive Statistics

	Percentiles	
	50th (Median)	75th
RespToPerf_33	4.0000	4.0000
RespToPerf_66	4.0000	4.0000
RespToPerf_100	4.0000	5.0000
RespToPerf_150	4.0000	4.0000
RespToPerf_200	4.0000	5.0000

Friedman Test

Ranks

	Mean Rank
RespToPerf_33	2.88
RespToPerf_66	2.91

RespToPerf_100	3.13
RespToPerf_150	3.07
RespToPerf_200	3.00

Test Statistics[a]

N	34
Chi-Square	1.033
df	4
Asymp. Sig.	.905

a. Friedman Test

```
NPAR TESTS
  /FRIEDMAN=Enjoyable_33 Enjoyable_66 Enjoyable_100 Enjoyable_150 Enjoyable_200
  /STATISTICS DESCRIPTIVES QUARTILES
  /MISSING LISTWISE.
```

NPar Tests

Notes

Output Created		19-APR-2014 21:54:05
Comments		
Input	Data	\\filer\home\My_Documents\encrypt7500632813994431560.spv.sav
	Active Dataset	DataSet1
	Filter	<none>
	Weight	<none>
	Split File	<none>
	N of Rows in Working Data File	34
Missing Value Handling	Definition of Missing	User-defined missing values are treated as missing.
	Cases Used	Statistics for all tests are based on cases with no missing data for any variables used.
Syntax		NPAR TESTS /FRIEDMAN=Enjoyable_33 Enjoyable_66 Enjoyable_100 Enjoyable_150 Enjoyable_200 /STATISTICS DESCRIPTIVES QUARTILES /MISSING LISTWISE.
Resources	Processor Time	00:00:00.02
	Elapsed Time	00:00:00.01
	Number of Cases Allowed[a]	78643

a. Based on availability of workspace memory.

Descriptive Statistics

	N	Mean	Std. Deviation	Minimum	Maximum	Percentiles
						25th
Enjoyable_33	34	3.1471	1.04830	1.00	5.00	3.0000
Enjoyable_66	34	3.1176	.91336	1.00	5.00	3.0000
Enjoyable_100	34	3.1765	.99911	1.00	5.00	2.0000
Enjoyable_150	34	2.9118	1.02596	1.00	5.00	2.0000
Enjoyable_200	34	2.9412	.98292	1.00	4.00	2.0000

Descriptive Statistics

	Percentiles	
	50th (Median)	75th
Enjoyable_33	3.0000	4.0000
Enjoyable_66	3.0000	4.0000
Enjoyable_100	3.0000	4.0000
Enjoyable_150	3.0000	4.0000
Enjoyable_200	3.0000	4.0000

Friedman Test

Ranks

	Mean Rank
Enjoyable_33	3.16
Enjoyable_66	3.09
Enjoyable_100	3.25
Enjoyable_150	2.75
Enjoyable_200	2.75

Test Statistics[a]

N	34
Chi-Square	5.565
df	4
Asymp. Sig.	.234

a. Friedman Test

```
EXAMINE VARIABLES=LATENCY_00 LATENCY_33 LATENCY_66 LATENCY_100
LATENCY_150 LATENCY_200
  /COMPARE VARIABLE
  /PLOT=BOXPLOT
  /STATISTICS=NONE
  /NOTOTAL
  /MISSING=LISTWISE.
```

```
EXAMINE VARIABLES=LATENCY_00 LATENCY_33 LATENCY_66 LATENCY_100
LATENCY_150 LATENCY_200
  /PLOT HISTOGRAM NPPLOT
  /STATISTICS DESCRIPTIVES
  /CINTERVAL 95
  /MISSING LISTWISE
  /NOTOTAL.
```

Explore

Notes

Output Created		04-MAY-2014 12:51:49
Comments		
Input	Data	\\filer\home\My_Documents\encrypt7500632813994431560.spv.sav
	Active Dataset	DataSet1
	Filter	<none>
	Weight	<none>
	Split File	<none>
	N of Rows in Working Data File	34
Missing Value Handling	Definition of Missing	User-defined missing values for dependent variables are treated as missing.
	Cases Used	Statistics are based on cases with no missing values for any dependent variable or factor used.
Syntax		EXAMINE VARIABLES=LATENCY_00 LATENCY_33 LATENCY_66 LATENCY_100 LATENCY_150 LATENCY_200 /PLOT HISTOGRAM NPPLOT /STATISTICS DESCRIPTIVES /CINTERVAL 95 /MISSING LISTWISE /NOTOTAL.
Resources	Processor Time	00:00:08.77
	Elapsed Time	00:00:07.54

Case Processing Summary

	Cases					
	Valid		Missing		Total	
	N	Percent	N	Percent	N	Percent
LATENCY_00	34	100.0%	0	0.0%	34	100.0%
LATENCY_33	34	100.0%	0	0.0%	34	100.0%
LATENCY_66	34	100.0%	0	0.0%	34	100.0%
LATENCY_100	34	100.0%	0	0.0%	34	100.0%
LATENCY_150	34	100.0%	0	0.0%	34	100.0%

LATENCY_200	34	100.0%	0	0.0%	34	100.0%

Descriptives

			Statistic	Std. Error
LATENCY_00	Mean		44.7888	4.47200
	95% Confidence Interval for Mean	Lower Bound	35.6905	
		Upper Bound	53.8872	
	5% Trimmed Mean		44.1856	
	Median		36.3400	
	Variance		679.959	
	Std. Deviation		26.07602	
	Minimum		7.69	
	Maximum		90.00	
	Range		82.31	
	Interquartile Range		41.29	
	Skewness		.777	.403
	Kurtosis		-.760	.788
LATENCY_33	Mean		28.9235	3.43706
	95% Confidence Interval for Mean	Lower Bound	21.9308	
		Upper Bound	35.9163	
	5% Trimmed Mean		26.4312	
	Median		21.5350	
	Variance		401.656	
	Std. Deviation		20.04135	
	Minimum		12.35	
	Maximum		90.00	
	Range		77.65	
	Interquartile Range		14.54	
	Skewness		2.163	.403
	Kurtosis		4.164	.788
LATENCY_66	Mean		31.5197	3.82574
	95% Confidence Interval for Mean	Lower Bound	23.7362	
		Upper Bound	39.3032	
	5% Trimmed Mean		29.3777	
	Median		24.0500	
	Variance		497.634	
	Std. Deviation		22.30771	
	Minimum		10.78	
	Maximum		90.00	
	Range		79.22	
	Interquartile Range		16.17	
	Skewness		1.717	.403
	Kurtosis		1.881	.788
LATENCY_100	Mean		40.8615	3.88232

	95% Confidence Interval for Mean	Lower Bound	32.9628	
		Upper Bound	48.7601	
	5% Trimmed Mean		39.5943	
	Median		32.3650	
	Variance		512.463	
	Std. Deviation		22.63764	
	Minimum		14.07	
	Maximum		90.00	
	Range		75.93	
	Interquartile Range		26.18	
	Skewness		1.172	.403
	Kurtosis		.389	.788
LATENCY_150	Mean		42.5762	3.57879
	95% Confidence Interval for Mean	Lower Bound	35.2951	
		Upper Bound	49.8573	
	5% Trimmed Mean		41.5672	
	Median		37.3200	
	Variance		435.462	
	Std. Deviation		20.86773	
	Minimum		11.09	
	Maximum		90.00	
	Range		78.91	
	Interquartile Range		26.92	
	Skewness		.854	.403
	Kurtosis		.373	.788
LATENCY_200	Mean		40.7803	3.35774
	95% Confidence Interval for Mean	Lower Bound	33.9489	
		Upper Bound	47.6117	
	5% Trimmed Mean		39.2819	
	Median		37.5500	
	Variance		383.330	
	Std. Deviation		19.57883	
	Minimum		17.21	
	Maximum		90.00	
	Range		72.79	
	Interquartile Range		24.48	
	Skewness		1.296	.403
	Kurtosis		1.464	.788

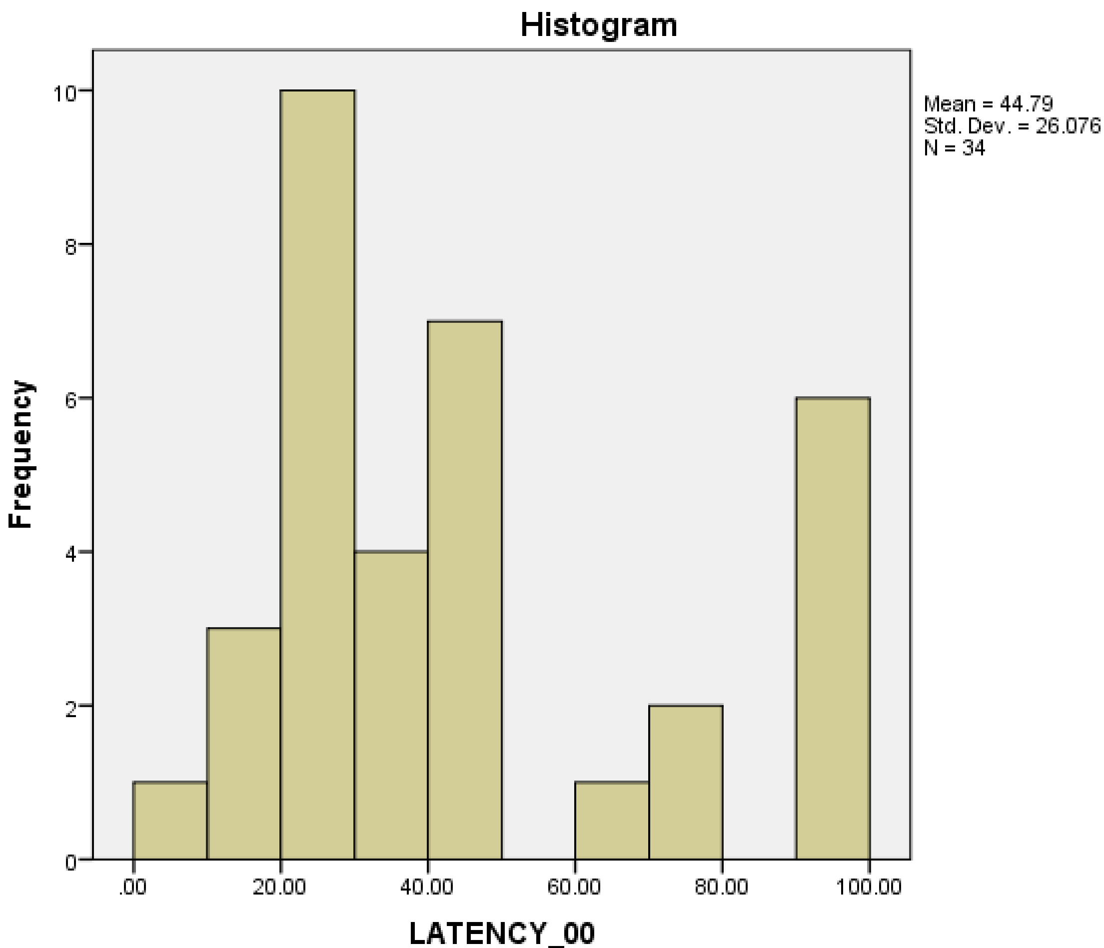
Tests of Normality

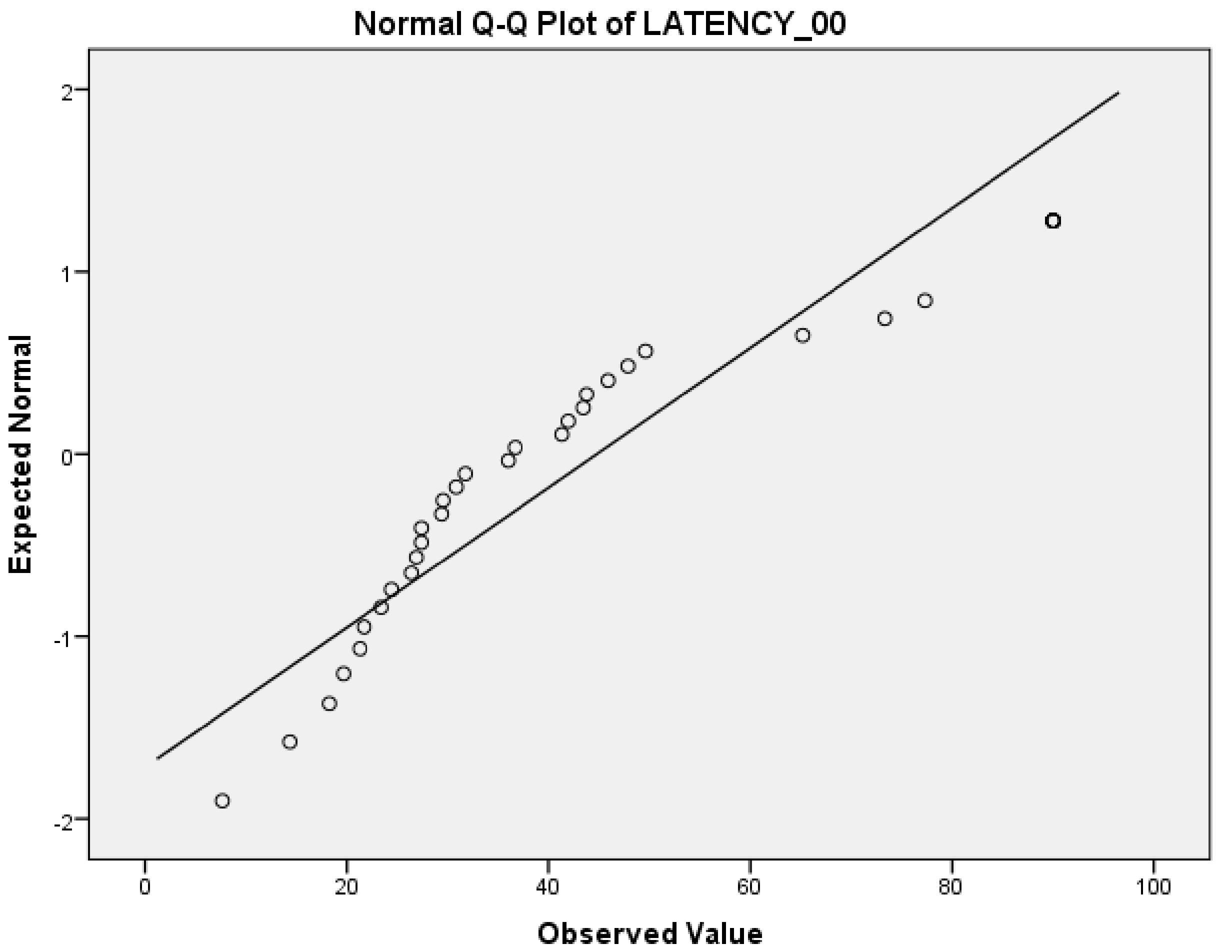
	Kolmogorov-Smirnov[a]			Shapiro-Wilk		
	Statistic	df	Sig.	Statistic	df	Sig.
LATENCY_00	.163	34	.023	.859	34	.000
LATENCY_33	.234	34	.000	.699	34	.000
LATENCY_66	.259	34	.000	.740	34	.000

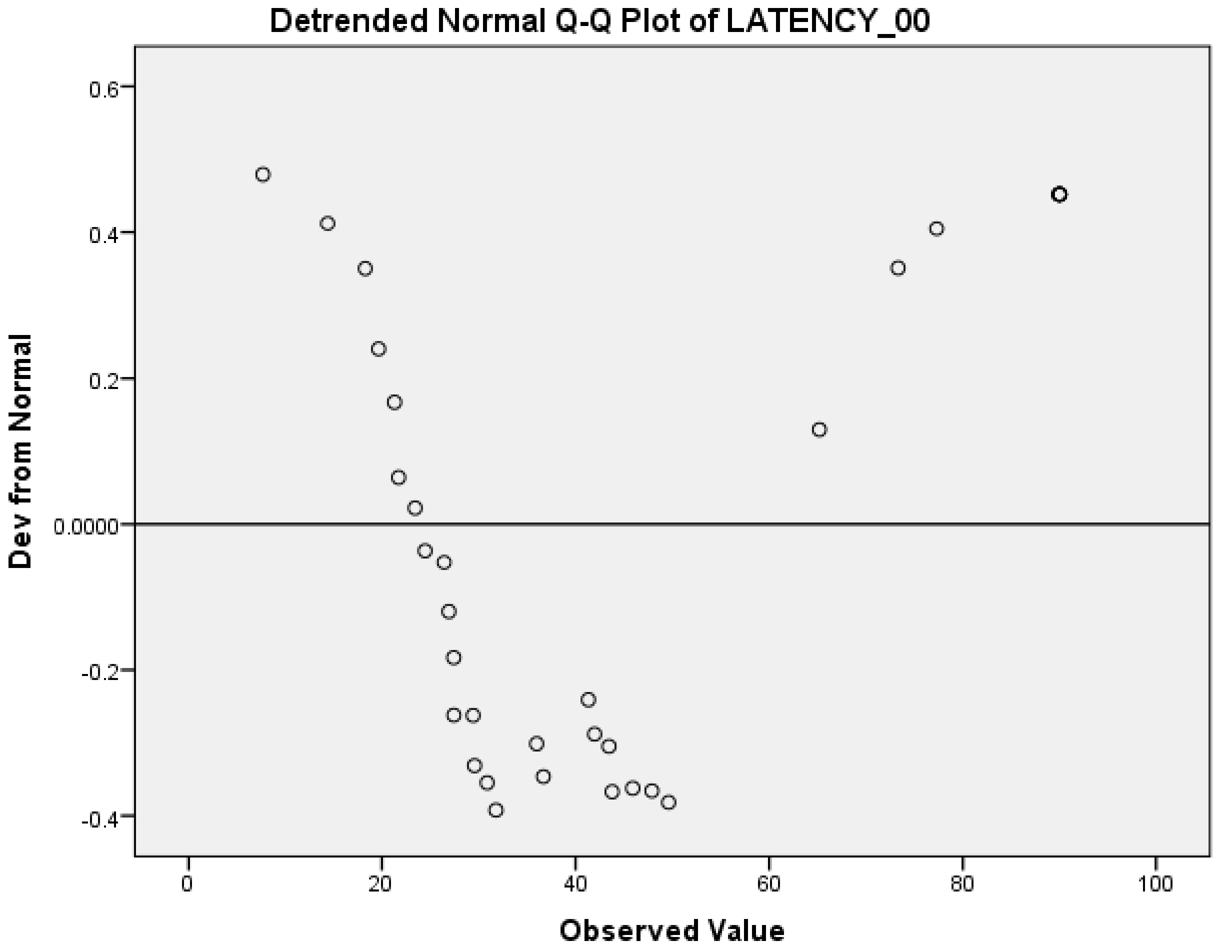
LATENCY_100	.200	34	.001	.843	34	.000
LATENCY_150	.127	34	.182	.927	34	.025
LATENCY_200	.120	34	.200*	.857	34	.000

*. This is a lower bound of the true significance.
a. Lilliefors Significance Correction

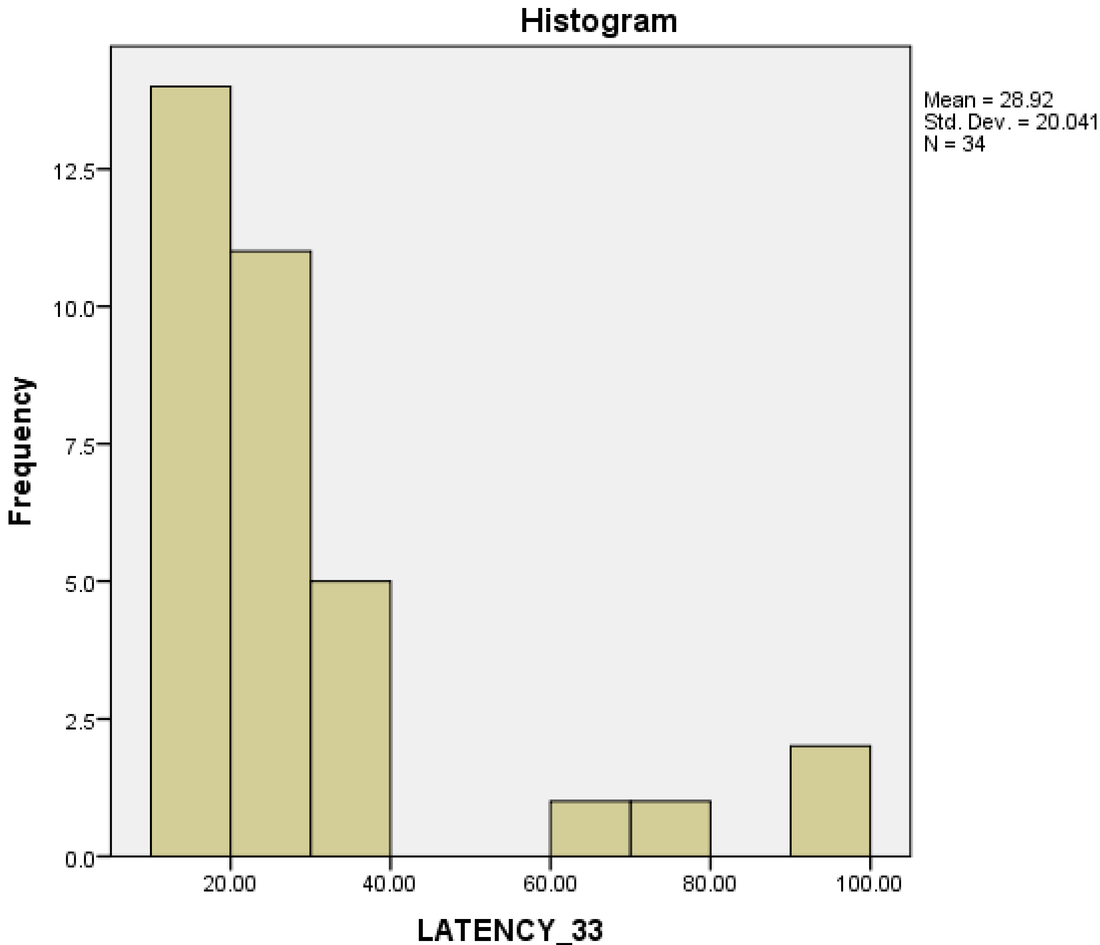
LATENCY_00

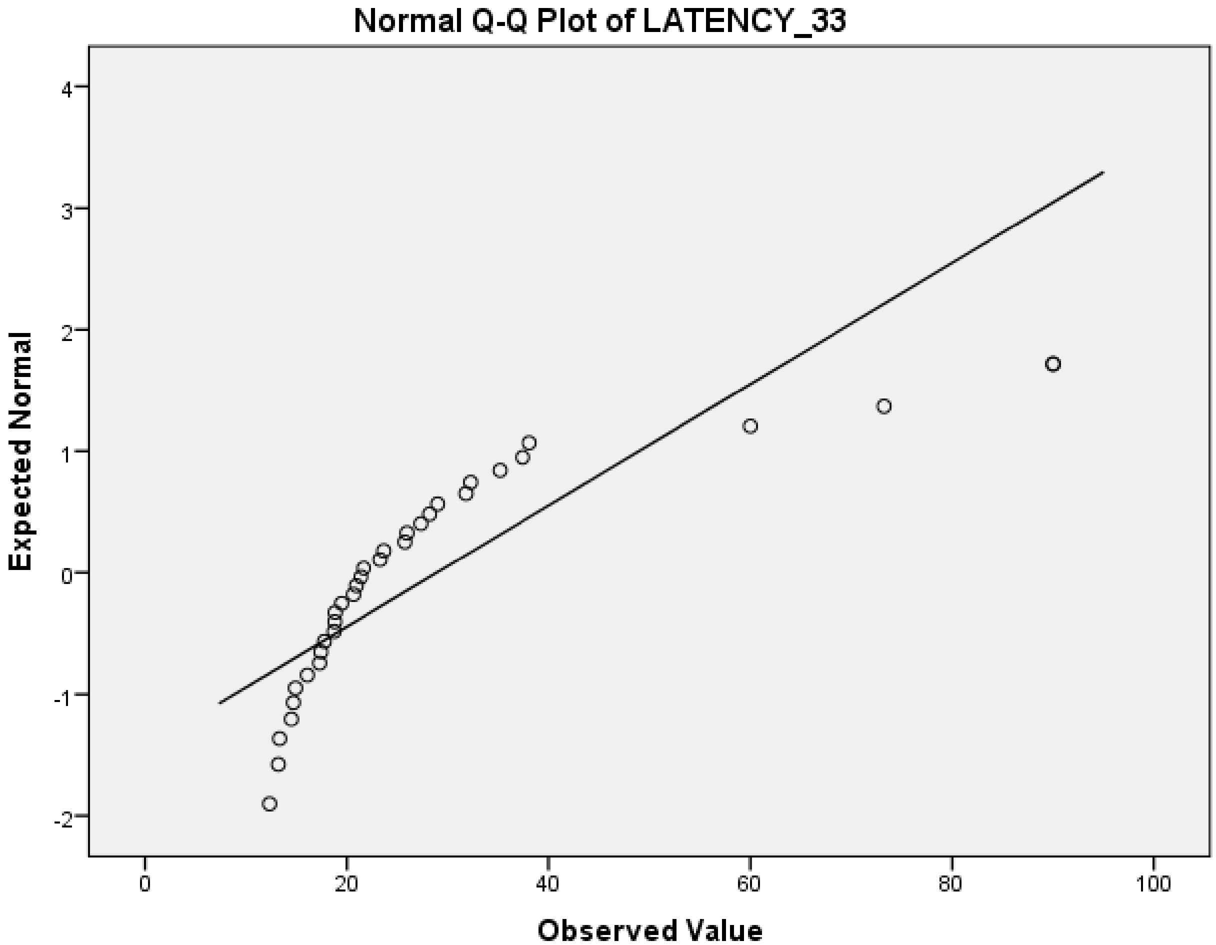


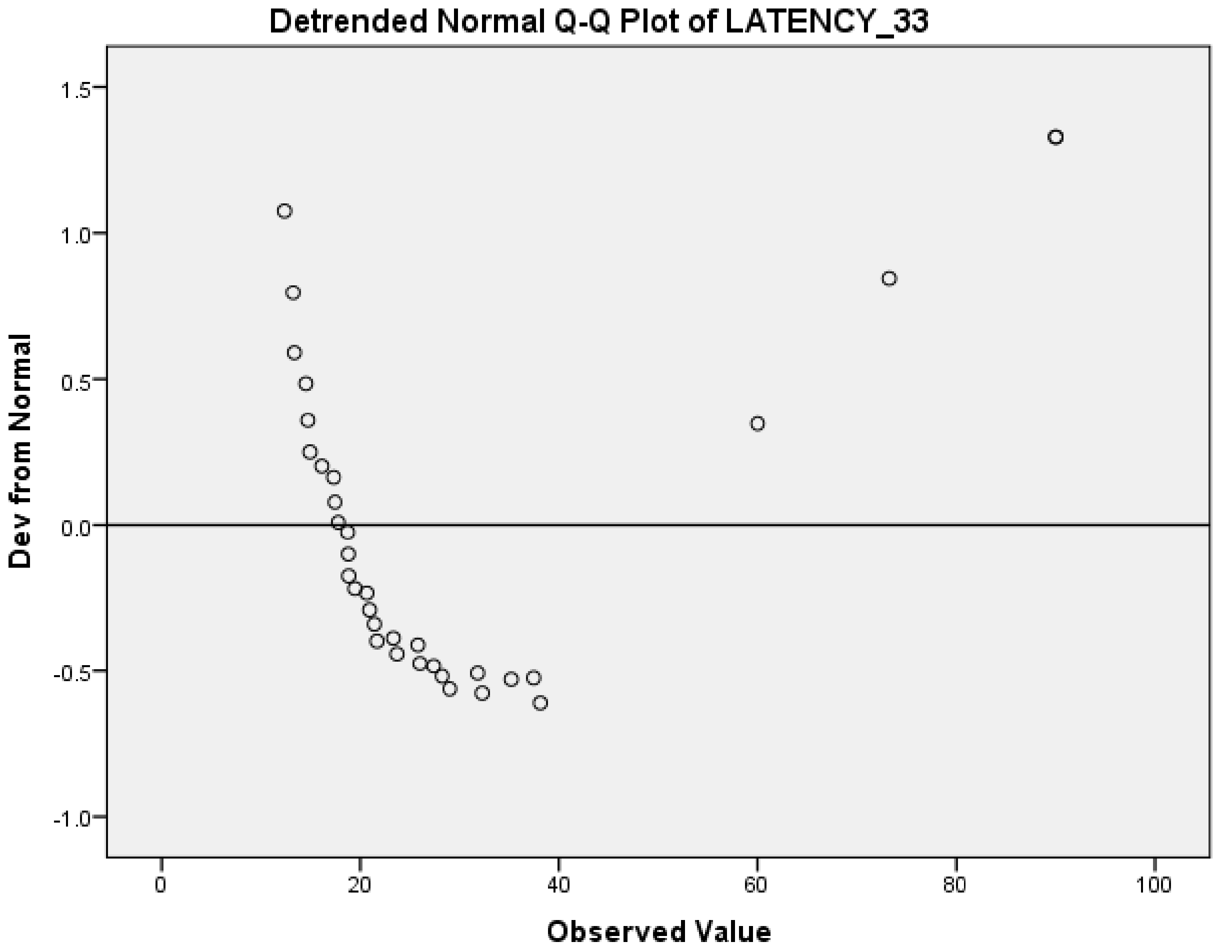

Normal Q-Q Plot of LATENCY_00
Expected Normal
2
1
0
-1
-2
0
20
40
60
80
100
Observed Value


Detrended Normal Q-Q Plot of LATENCY_00
Dev from Normal
Observed Value
0.6
0.4
0.2
0.0000
-0.2
-0.4
0
20
40
60
80
100

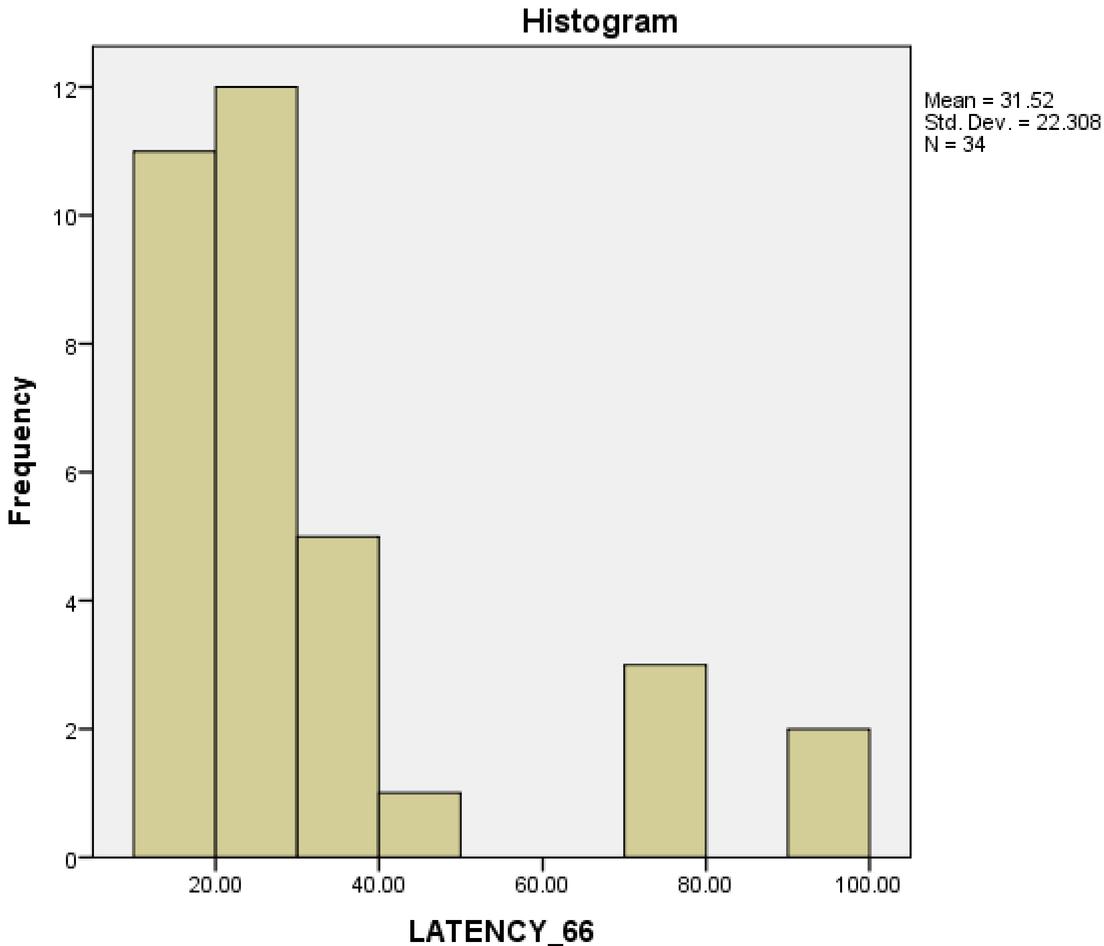
LATENCY_33

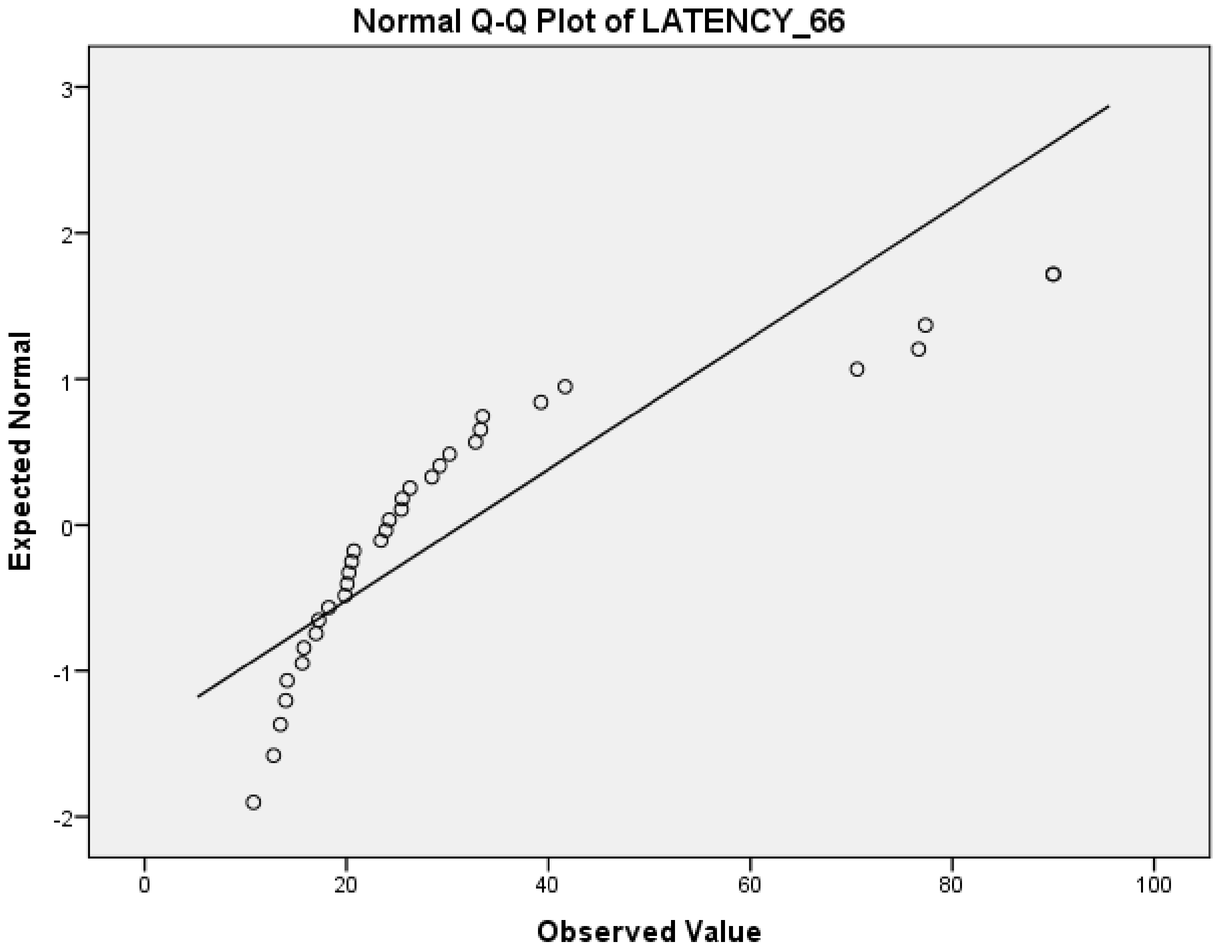

Histogram
Mean = 28.92
Std. Dev. = 20.041
N = 34
Frequency
12.5
10.0
7.5
5.0
2.5
0.0
20.00
40.00
60.00
80.00
100.00
LATENCY_33

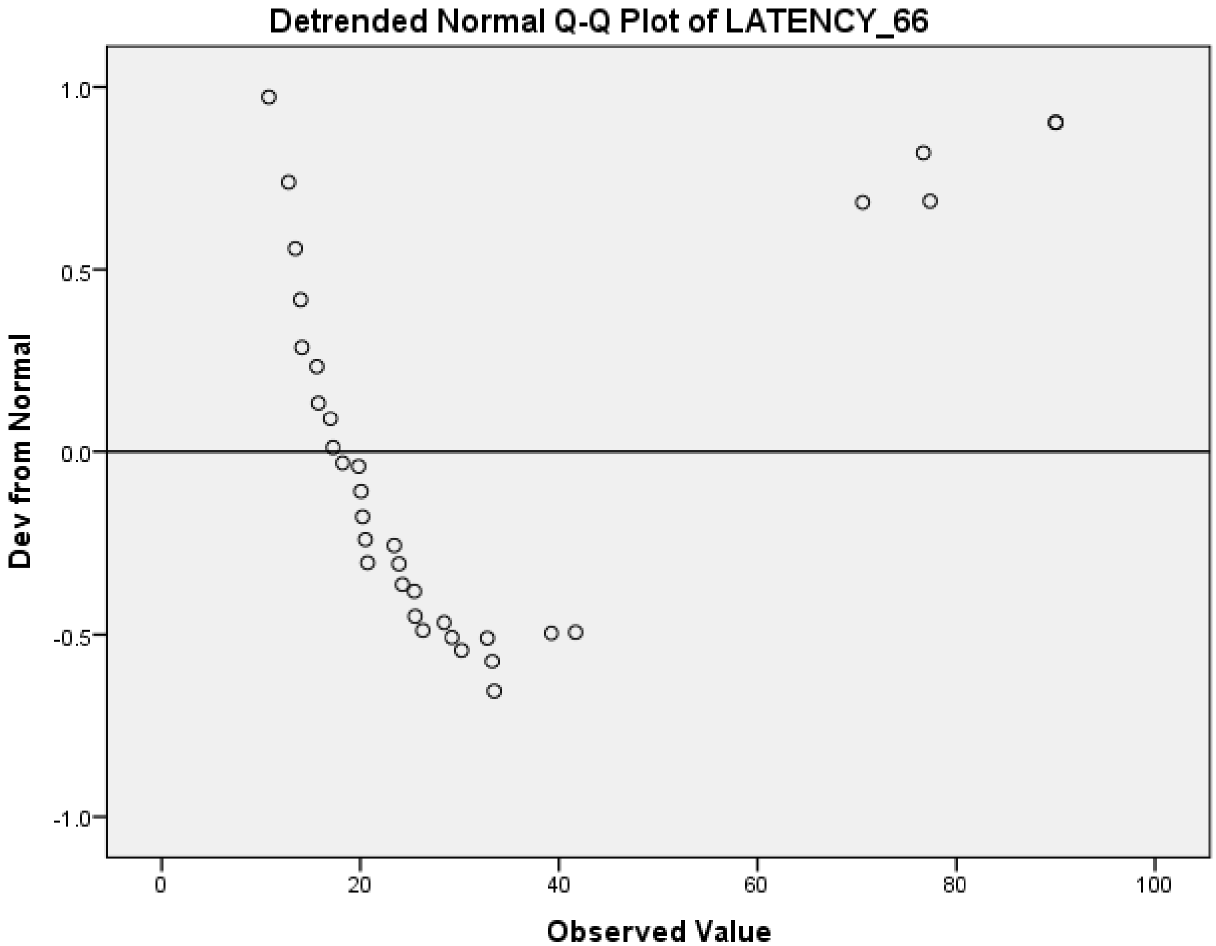

Normal Q-Q Plot of LATENCY_33
Expected Normal
Observed Value
4
3
2
1
0
-1
-2
0
20
40
60
80
100


Detrended Normal Q-Q Plot of LATENCY_33
Dev from Normal
1.5
1.0
0.5
0.0
-0.5
-1.0
0
20
40
60
80
100
Observed Value

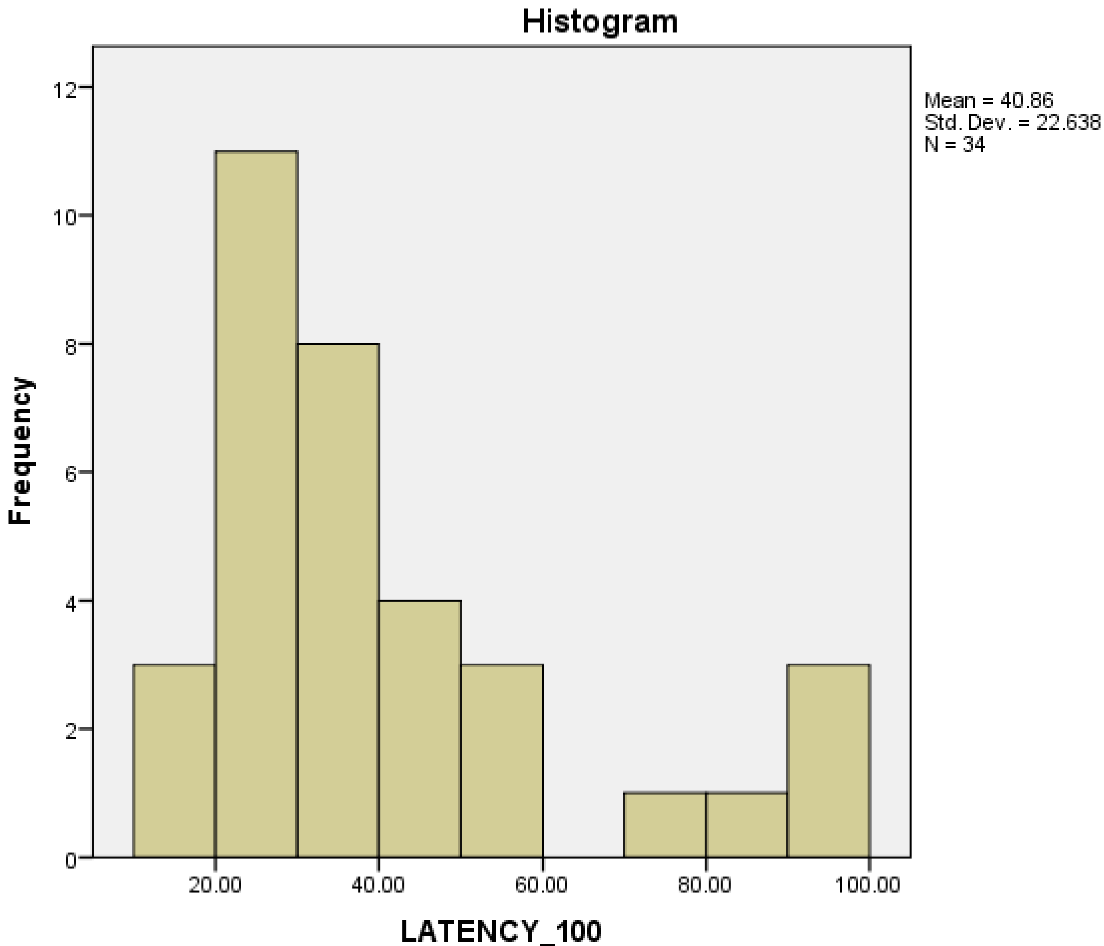
LATENCY_66

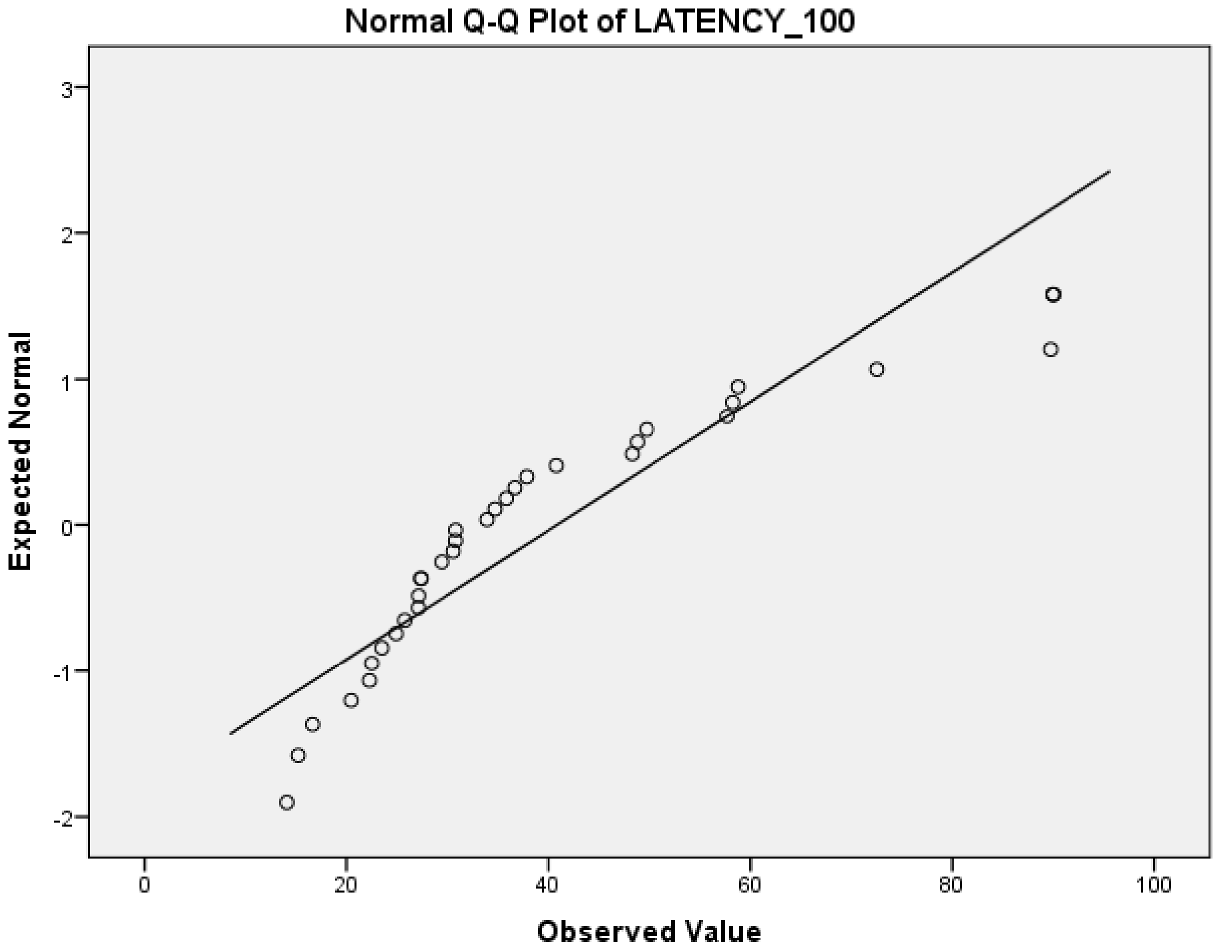

Histogram
Mean = 31.52
Std. Dev. = 22.308
N = 34
Frequency
0
2
4
6
8
10
12
20.00
40.00
60.00
80.00
100.00
LATENCY_66


Normal Q-Q Plot of LATENCY_66
Expected Normal
Observed Value
3
2
1
0
-1
-2
0
20
40
60
80
100

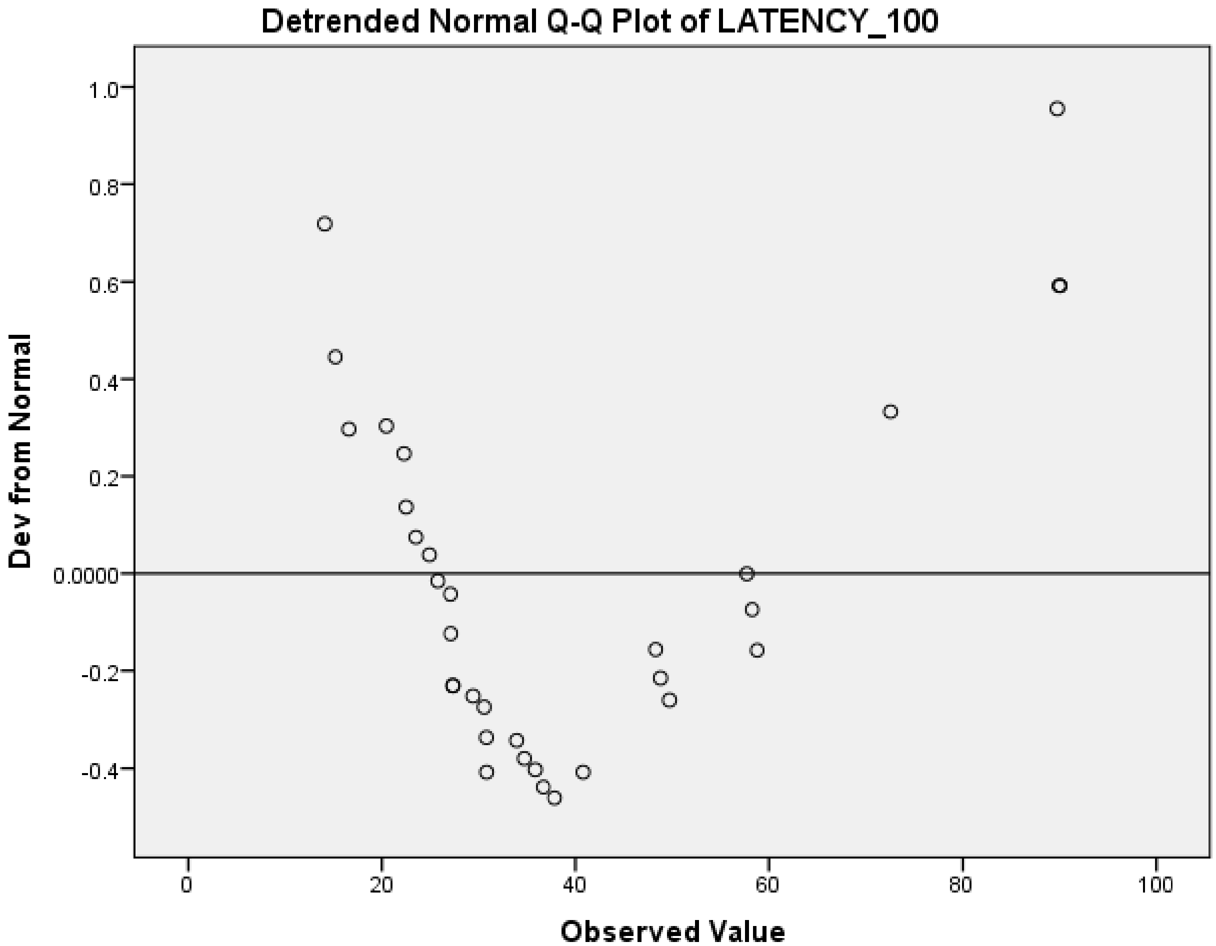

Detrended Normal Q-Q Plot of LATENCY_66
Dev from Normal
1.0
0.5
0.0
-0.5
-1.0
0
20
40
60
80
100
Observed Value

LATENCY_100

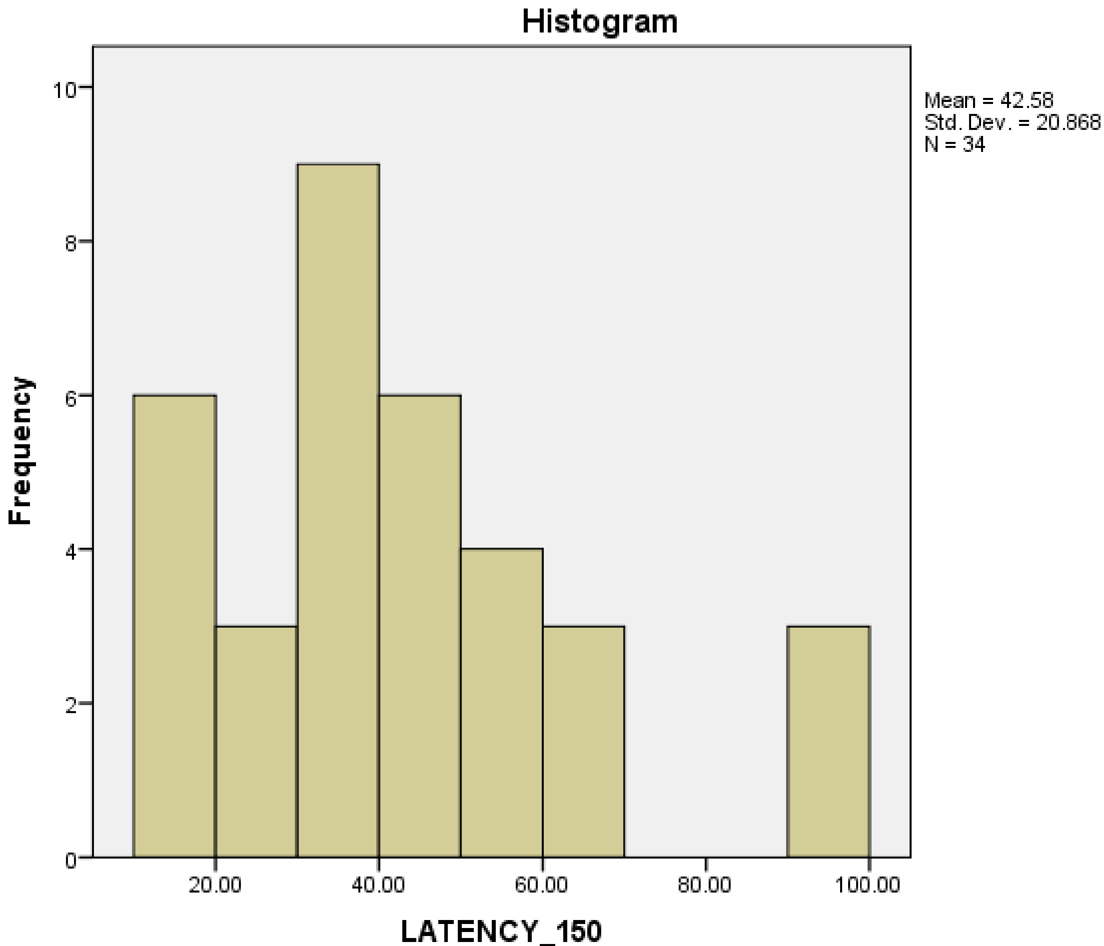

Histogram
Mean = 40.86
Std. Dev. = 22.638
N = 34
Frequency
0
2
4
6
8
10
12
20.00
40.00
60.00
80.00
100.00
LATENCY_100

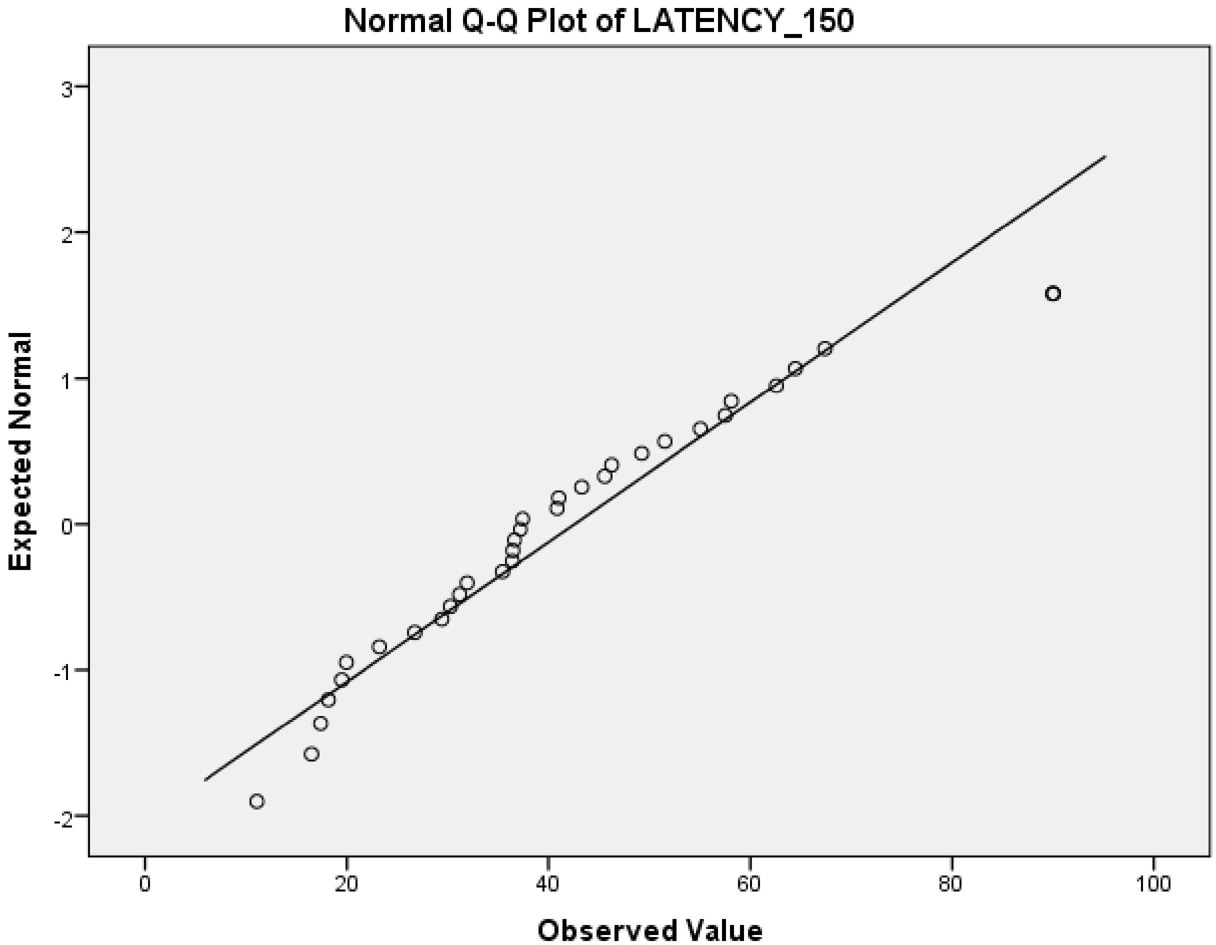


Normal Q-Q Plot of LATENCY_100
Expected Normal
Observed Value
3
2
1
0
-1
-2
0
20
40
60
80
100

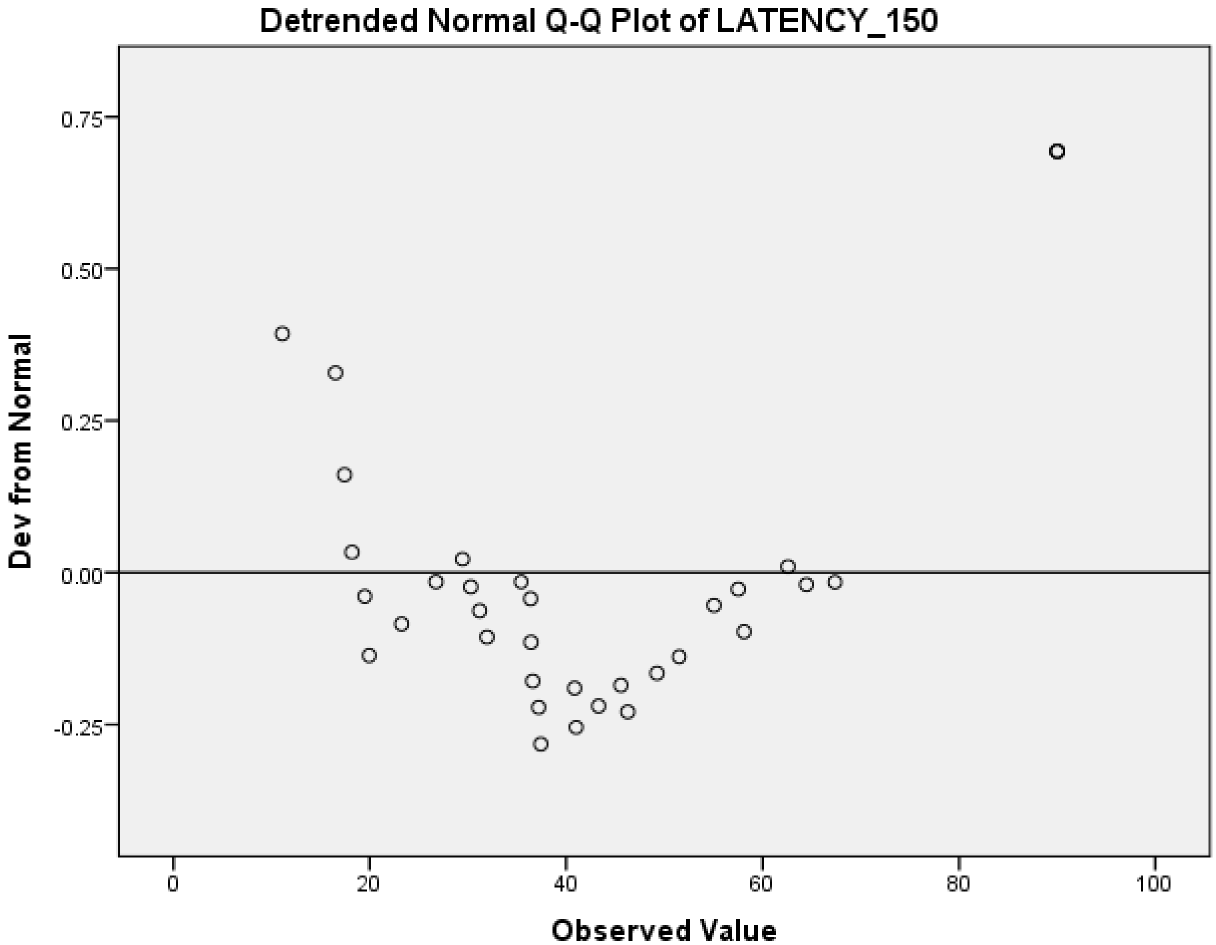

Detrended Normal Q-Q Plot of LATENCY_100
Dev from Normal
1.0
0.8
0.6
0.4
0.2
0.0000
-0.2
-0.4
0
20
40
60
80
100
Observed Value

LATENCY_150


Histogram
Mean = 42.58
Std. Dev. = 20.868
N = 34
Frequency
0
2
4
6
8
10
20.00
40.00
60.00
80.00
100.00
LATENCY_150

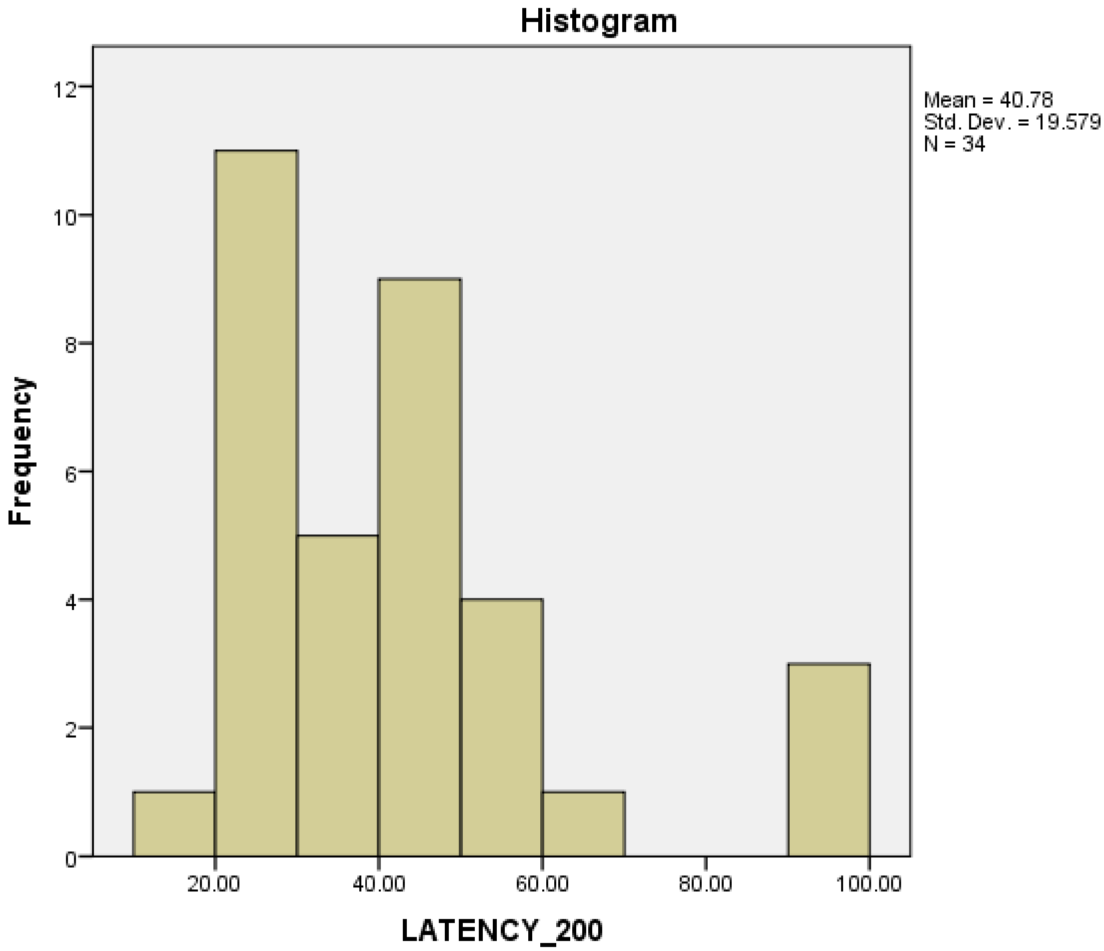


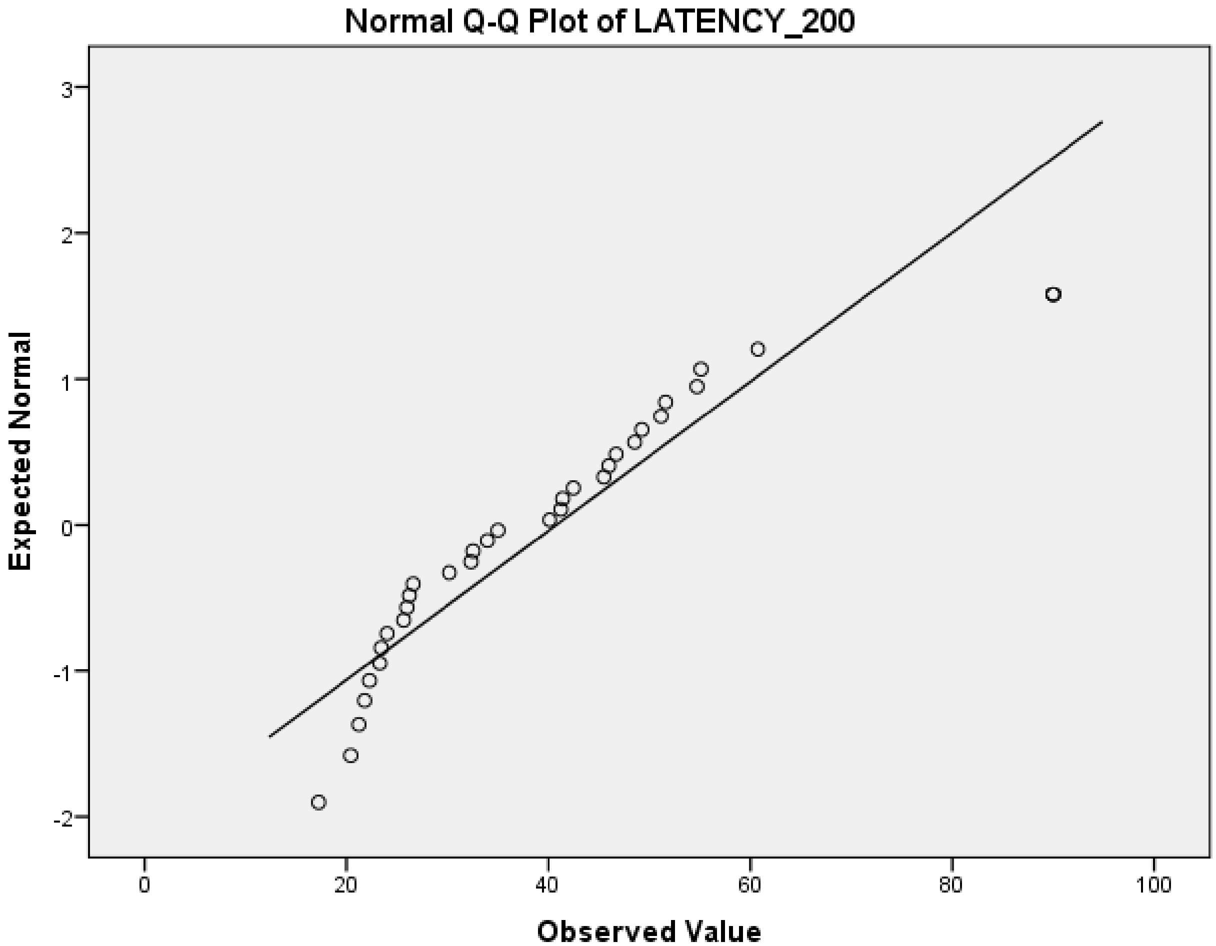
Normal Q-Q Plot of LATENCY_150
Expected Normal
3
2
1
0
-1
-2
0
20
40
60
80
100
Observed Value



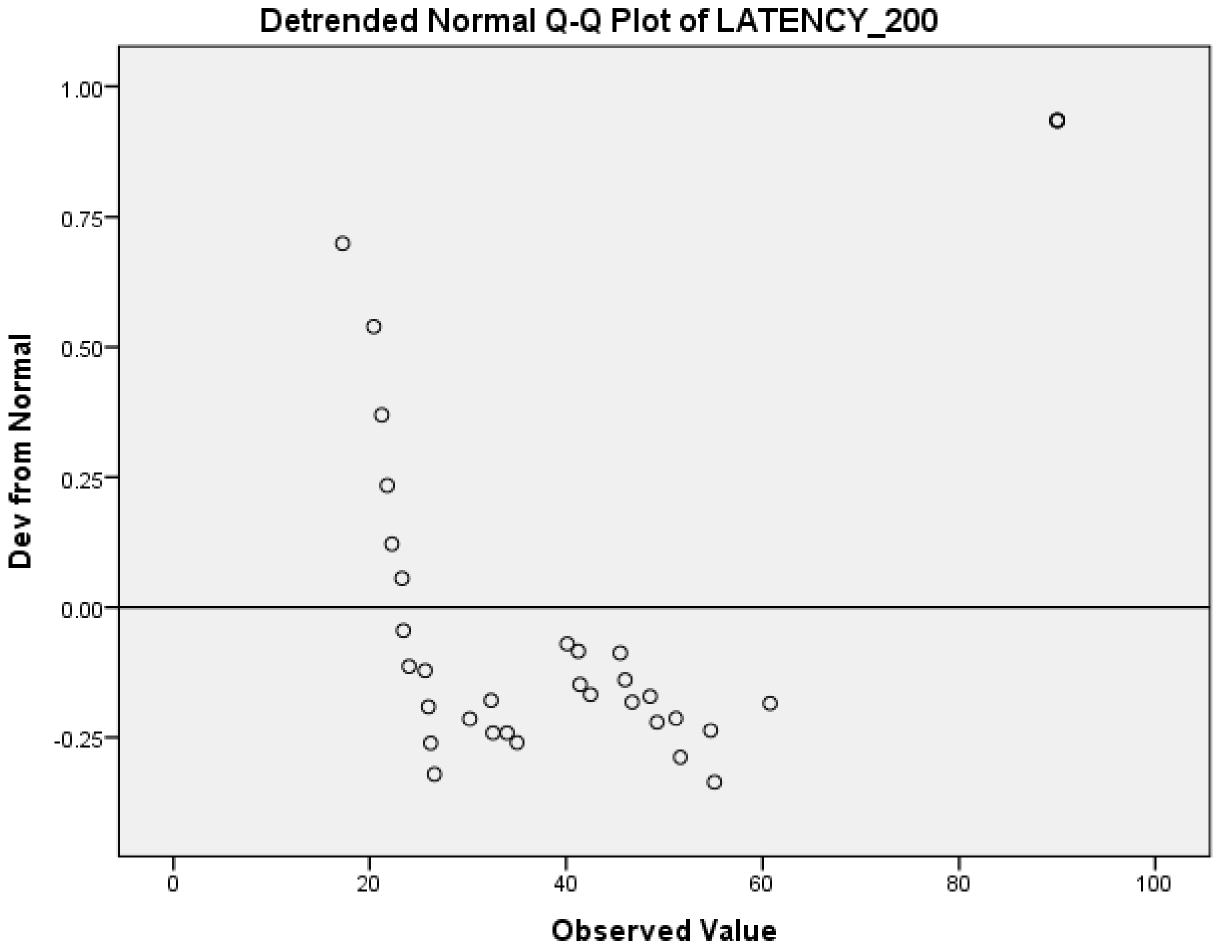
Detrended Normal Q-Q Plot of LATENCY_150
Dev from Normal
0.75
0.50
0.25
0.00
-0.25
0
20
40
60
80
100
Observed Value

LATENCY_200


Histogram
Mean = 40.78
Std. Dev. = 19.579
N = 34
Frequency
0
2
4
6
8
10
12
20.00
40.00
60.00
80.00
100.00
LATENCY_200



Normal Q-Q Plot of LATENCY_200
Expected Normal
Observed Value
3
2
1
0
-1
-2
0
20
40
60
80
100




```
DATASET ACTIVATE DataSet1.

SAVE OUTFILE='\\filer\home\My_Documents\encrypt7500632813994431560.spv.sav'
  /COMPRESSED.
GLM LATENCY_33 LATENCY_66 LATENCY_100 LATENCY_150 LATENCY_200
  /WSFACTOR=Latency 5 Polynomial
  /MEASURE=TimeToComplete
  /METHOD=SSTYPE(3)
  /EMMEANS=TABLES(Latency) COMPARE ADJ(BONFERRONI)
  /PRINT=DESCRIPTIVE ETASQ HOMOGENEITY
  /CRITERIA=ALPHA(.05)
  /WSDESIGN=Latency.


GLM LATENCY_33 LATENCY_66 LATENCY_100 LATENCY_150 LATENCY_200
  /WSFACTOR=Latency 5 Polynomial
  /MEASURE=TimeToComplete
```

```
/METHOD=SSTYPE(3)
/PLOT=PROFILE(Latency)
/EMMEANS=TABLES(Latency) COMPARE ADJ(BONFERRONI)
/PRINT=DESCRIPTIVE ETASQ HOMOGENEITY
/CRITERIA=ALPHA(.05)
/WSDESIGN=Latency.
```

General Linear Model

Notes

Output Created Comments		04-MAY-2014 13:02:53
Input	Data	\\filer\home\My_Documents\encrypt7500632813994431560.spv.sav
	Active Dataset	DataSet1
	Filter	<none>
	Weight	<none>
	Split File	<none>
	N of Rows in Working Data File	34
Missing Value Handling	Definition of Missing	User-defined missing values are treated as missing.
	Cases Used	Statistics are based on all cases with valid data for all variables in the model.
Syntax		GLM LATENCY_33 LATENCY_66 LATENCY_100 LATENCY_150 LATENCY_200 /WSFACTOR=Latency 5 Polynomial /MEASURE=TimeToComplete /METHOD=SSTYPE(3) /PLOT=PROFILE(Latency) /EMMEANS=TABLES(Latency) COMPARE ADJ(BONFERRONI) /PRINT=DESCRIPTIVE ETASQ HOMOGENEITY /CRITERIA=ALPHA(.05) /WSDESIGN=Latency.
Resources	Processor Time	00:00:00.28
	Elapsed Time	00:00:00.27

Warnings

The HOMOGENEITY specification in the PRINT subcommand will be ignored because there are no between-subjects factors.

Within-Subjects Factors

Measure: TimeToComplete

Latency	Dependent Variable
1	LATENCY_33
2	LATENCY_66
3	LATENCY_100
4	LATENCY_150
5	LATENCY_200

Descriptive Statistics

	Mean	Std. Deviation	N
LATENCY_33	28.9235	20.04135	34
LATENCY_66	31.5197	22.30771	34
LATENCY_100	40.8615	22.63764	34
LATENCY_150	42.5762	20.86773	34
LATENCY_200	40.7803	19.57883	34

Multivariate Tests[a]

Effect		Value	F	Hypothesis df	Error df	Sig.
Latency	Pillai's Trace	.545	8.983[b]	4.000	30.000	.000
	Wilks' Lambda	.455	8.983[b]	4.000	30.000	.000
	Hotelling's Trace	1.198	8.983[b]	4.000	30.000	.000
	Roy's Largest Root	1.198	8.983[b]	4.000	30.000	.000

Multivariate Tests[a]

Effect		Partial Eta Squared
Latency	Pillai's Trace	.545
	Wilks' Lambda	.545
	Hotelling's Trace	.545
	Roy's Largest Root	.545

a. Design: Intercept
Within Subjects Design: Latency
b. Exact statistic

Mauchly's Test of Sphericity[a]

Measure: TimeToComplete

Within Subjects Effect	Mauchly's W	Approx. Chi-Square	df	Sig.	Epsilon[b] Greenhouse-Geisser
Latency	.498	21.908	9	.009	.789

Mauchly's Test of Sphericity[a]

Measure: TimeToComplete

Within Subjects Effect	Epsilon	
	Huynh-Feldt	Lower-bound
Latency	.882	.250

Tests the null hypothesis that the error covariance matrix of the orthonormalized transformed dependent variables is proportional to an identity matrix.[a]
a. Design: Intercept
Within Subjects Design: Latency

b. May be used to adjust the degrees of freedom for the averaged tests of significance. Corrected tests are displayed in the Tests of Within-Subjects Effects table.

Tests of Within-Subjects Effects

Measure: TimeToComplete

Source		Type III Sum of Squares	df	Mean Square	F
Latency	Sphericity Assumed	5288.202	4	1322.051	5.365
	Greenhouse-Geisser	5288.202	3.155	1676.105	5.365
	Huynh-Feldt	5288.202	3.527	1499.225	5.365
	Lower-bound	5288.202	1.000	5288.202	5.365
Error(Latency)	Sphericity Assumed	32526.117	132	246.410	
	Greenhouse-Geisser	32526.117	104.117	312.400	
	Huynh-Feldt	32526.117	116.401	279.433	
	Lower-bound	32526.117	33.000	985.640	

Tests of Within-Subjects Effects

Measure: TimeToComplete

Source		Sig.	Partial Eta Squared
Latency	Sphericity Assumed	.000	.140
	Greenhouse-Geisser	.001	.140
	Huynh-Feldt	.001	.140
	Lower-bound	.027	.140
Error(Latency)	Sphericity Assumed		
	Greenhouse-Geisser		
	Huynh-Feldt		
	Lower-bound		

Tests of Within-Subjects Contrasts

Measure: TimeToComplete

Source	Latency	Type III Sum of Squares	df	Mean Square	F	Sig.
Latency	Linear	4110.440	1	4110.440	15.675	.000
	Quadratic	654.079	1	654.079	1.985	.168
	Cubic	357.643	1	357.643	2.920	.097
	Order 4	166.040	1	166.040	.612	.440
Error(Latency)	Linear	8653.667	33	262.232		
	Quadratic	10872.743	33	329.477		
	Cubic	4041.254	33	122.462		
	Order 4	8958.453	33	271.468		

Tests of Within-Subjects Contrasts

Measure: TimeToComplete

Source	Latency	Partial Eta Squared
Latency	Linear	.322
	Quadratic	.057
	Cubic	.081
	Order 4	.018

Error(Latency)	Linear	
	Quadratic	
	Cubic	
	Order 4	

Tests of Between-Subjects Effects

Measure: TimeToComplete
Transformed Variable: Average

Source	Type III Sum of Squares	df	Mean Square	F	Sig.	Partial Eta Squared
Intercept	231878.301	1	231878.301	186.262	.000	.849
Error	41081.865	33	1244.905			

Estimated Marginal Means

Latency

Estimates

Measure: TimeToComplete

Latency	Mean	Std. Error	95% Confidence Interval	
			Lower Bound	Upper Bound
1	28.924	3.437	21.931	35.916
2	31.520	3.826	23.736	39.303
3	40.861	3.882	32.963	48.760
4	42.576	3.579	35.295	49.857
5	40.780	3.358	33.949	47.612

Pairwise Comparisons

Measure: TimeToComplete

(I) Latency	(J) Latency	Mean Difference (I-J)	Std. Error	Sig.[b]	95% Confidence Interval for Difference[b]	
					Lower Bound	Upper Bound
1	2	-2.596	3.960	1.000	-14.509	9.316
	3	-11.938*	3.337	.011	-21.976	-1.900
	4	-13.653*	4.246	.029	-26.426	-.880
	5	-11.857*	3.626	.025	-22.764	-.949
2	1	2.596	3.960	1.000	-9.316	14.509
	3	-9.342*	2.936	.032	-18.174	-.510
	4	-11.056*	3.079	.011	-20.320	-1.793
	5	-9.261	4.516	.483	-22.847	4.326
3	1	11.938*	3.337	.011	1.900	21.976

	2	9.342*	2.936	.032	.510	18.174
	4	-1.715	4.106	1.000	-14.067	10.638
	5	.081	4.047	1.000	-12.094	12.256
4	1	13.653*	4.246	.029	.880	26.426
	2	11.056*	3.079	.011	1.793	20.320
	3	1.715	4.106	1.000	-10.638	14.067
	5	1.796	3.901	1.000	-9.939	13.530
5	1	11.857*	3.626	.025	.949	22.764
	2	9.261	4.516	.483	-4.326	22.847
	3	-.081	4.047	1.000	-12.256	12.094
	4	-1.796	3.901	1.000	-13.530	9.939

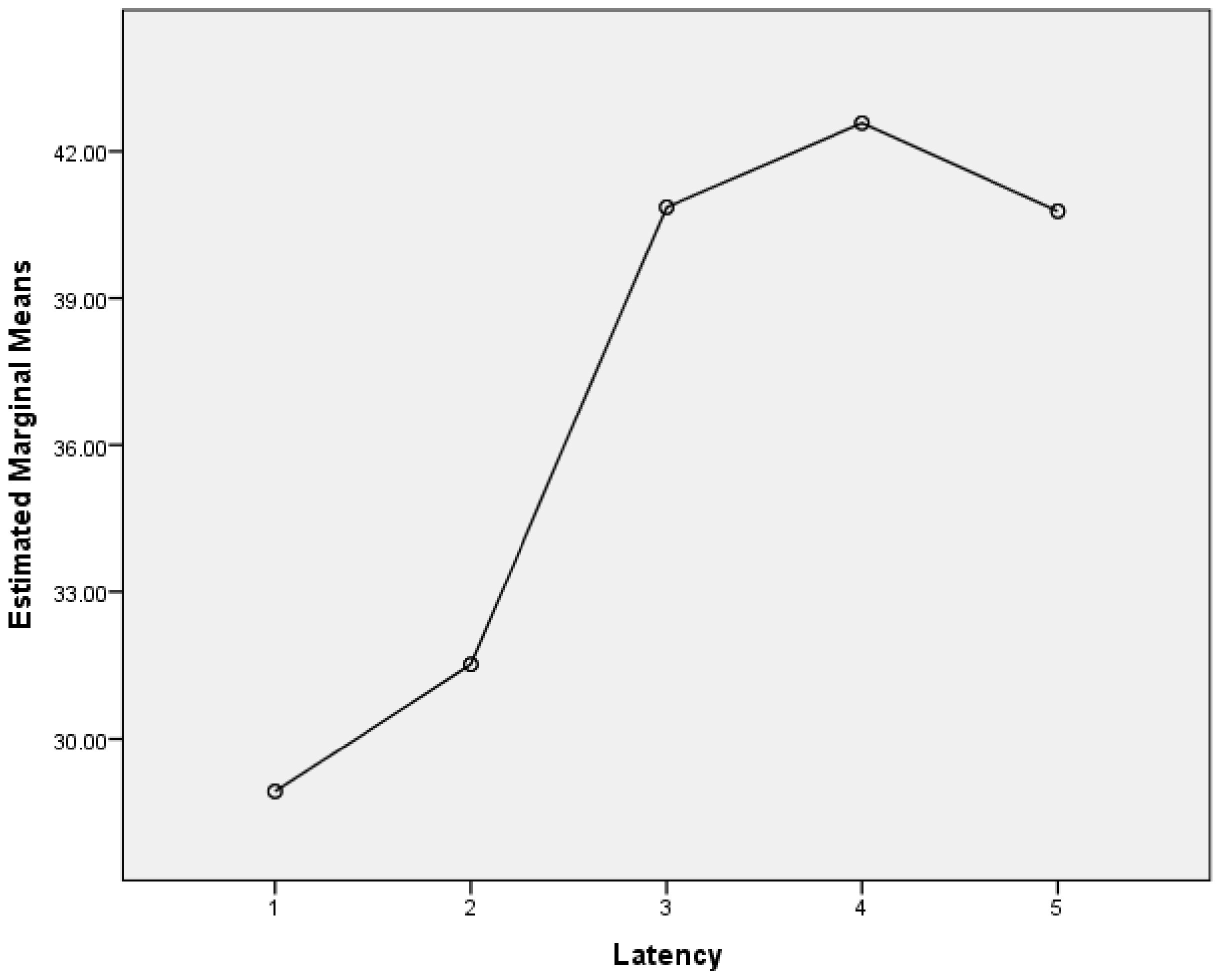
Based on estimated marginal means
*. The mean difference is significant at the .05 level.
b. Adjustment for multiple comparisons: Bonferroni.

Multivariate Tests

	Value	F	Hypothesis df	Error df	Sig.	Partial Eta Squared
Pillai's trace	.545	8.983[a]	4.000	30.000	.000	.545
Wilks' lambda	.455	8.983[a]	4.000	30.000	.000	.545
Hotelling's trace	1.198	8.983[a]	4.000	30.000	.000	.545
Roy's largest root	1.198	8.983[a]	4.000	30.000	.000	.545

Each F tests the multivariate effect of Latency. These tests are based on the linearly independent pairwise comparisons among the estimated marginal means.
a. Exact statistic

Profile Plots



NPAR TESTS
  /FRIEDMAN=QOE_33 QOE_66 QOE_100 QOE_150 QOE_200
  /STATISTICS DESCRIPTIVES
  /MISSING LISTWISE.

EXAMINE VARIABLES=LATENCY_00 LATENCY_33 LATENCY_66 LATENCY_100 LATENCY_150 LATENCY_200
  /COMPARE VARIABLE
  /PLOT=BOXPLOT
  /STATISTICS=NONE
  /NOTOTAL
  /MISSING=LISTWISE.

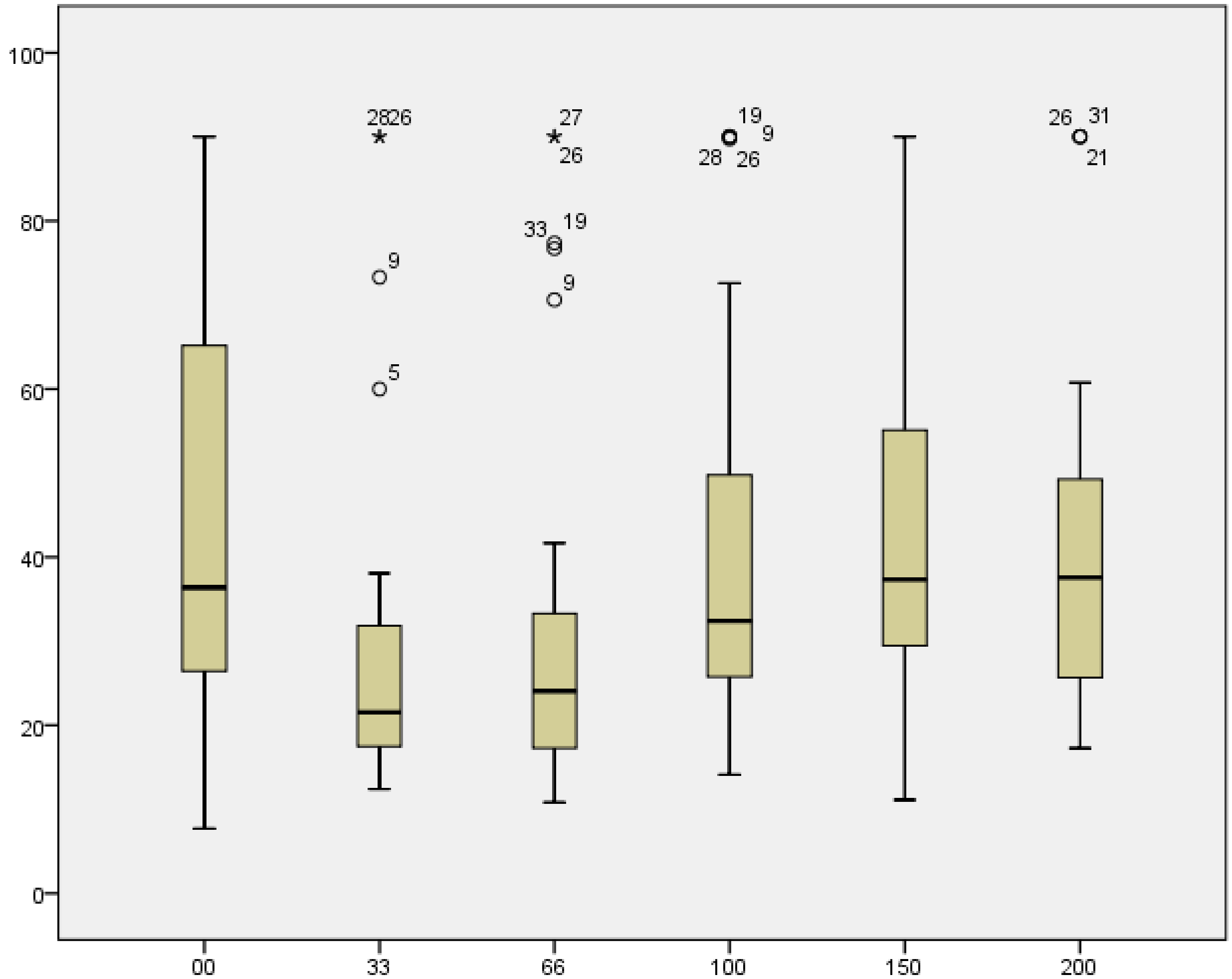
Explore

Notes

Output Created		04-MAY-2014 13:17:20
Comments		
Input	Data	\\filer\home\My_Documents\encrypt7500632813994431560.spv.sav
	Active Dataset	DataSet1
	Filter	<none>
	Weight	<none>
	Split File	<none>
	N of Rows in Working Data File	34
Missing Value Handling	Definition of Missing	User-defined missing values for dependent variables are treated as missing.
	Cases Used	Statistics are based on cases with no missing values for any dependent variable or factor used.
Syntax		EXAMINE VARIABLES=LATENCY_00 LATENCY_33 LATENCY_66 LATENCY_100 LATENCY_150 LATENCY_200 /COMPARE VARIABLE /PLOT=BOXPLOT /STATISTICS=NONE /NOTOTAL /MISSING=LISTWISE.
Resources	Processor Time	00:00:00.28
	Elapsed Time	00:00:00.68

Case Processing Summary

	Cases					
	Valid		Missing		Total	
	N	Percent	N	Percent	N	Percent
LATENCY_00	34	100.0%	0	0.0%	34	100.0%
LATENCY_33	34	100.0%	0	0.0%	34	100.0%
LATENCY_66	34	100.0%	0	0.0%	34	100.0%
LATENCY_100	34	100.0%	0	0.0%	34	100.0%
LATENCY_150	34	100.0%	0	0.0%	34	100.0%
LATENCY_200	34	100.0%	0	0.0%	34	100.0%



100
80
60
40
20
0
28 26
9
5
27
26
33
19
9
19
9
28
26
26
31
21
00
33
66
100
150
200