Project 2 Report Please do not write your name for

STA 315 W21 anonymous grading

Instructions: As in Project 1, answer the following questions about your experiment. Please leave the original questions in the document.

General Question

1. [1] What is the general question you seek to answer?

On my laptop, how can I get the best synthetic benchmark scores for my CPU depending on the voltage allowed to my CPU (this is measured in an offset based on the standard voltage to the CPU) and clock speeds for the CPU (3.3,3.4 ect).

1. [1] What is your motivation for studying this question?  
   My motivation for studying this is I love playing with computers and trying to overclock (OC) and maximize performance on my laptop.

3 Choices: responses, conditions, material

1. [2] Identify your response.

I’ll be using one run (test) of the synthetic benchmarks ‘Cinebench R23’ to objectively measure CPU performance. (ex. a score of 6363)

* + [1] What 2 factors did you study and what are the levels of each factor?   
     The factors of interest that I am going to be investigating in this study is going to **voltage to the CPU** (measured in an offset as I will actually be ‘down-volting’ the CPU by giving it less power (ex. 0 being normal settings, -80.1 and -160.2) so it wont over heat which should give extra performance to the CPU) and the **clock speed** of the CPU (3.4, 3.5 GHz ect)
  + [1] Identify each of these factors as an **observational factor** or an **experimental factor**. For an observational factor, the levels are “built into” the units, like the conditions in an observational study; for an experimental factor, the levels are assigned to the units, like the conditions in an experiment. **It is required that at least one of the factors be an experimental factor**.

Both factors in my experiment are going to be experimental factors as I am going to be artificially setting the speed of the CPU clock speed and the number for the offset voltage, which will be negative since I’m going to be down-volting my CPU

* + [1] What are the treatment combinations? Replace the entries of the table below with the factors, levels, and treatments in your experiment (adding rows and/or columns for more levels if necessary):

|  |  |  |  |
| --- | --- | --- | --- |
| Offset Voltage (mV) of CPU | CPU Clock Speed (GHz) | |  |
| 3.3 | 3.4 | 3.5 |
| 0 (essentially default voltage) | 0, 3.3 | 0, 3.4 | 0, 3.5 |
| -80.1 | -80.1, 3.3 | -80.1, 3.4 | -80.1, 3.5 |
| -160.2 | -160.2, 3.3 | -160.2, 3.4 | -160.2, 3.5 |

1. [2] Identify the experimental units. (This is what you assign the treatments to.)   
     
   In this case, the experimental unit would be one run of the program Cinebench R23
2. [1] What is the specific question you seek to answer?(This comes from applying your choices of the response, conditions, and the material to the more general question you posed at the beginning.)  
   ­­­  
   How is the performance of the CPU, measured through the synthetic benchmark ‘Cinebench R23’, affected by its core clock speed and voltage offset when the fan speed is constant.
3. [1] It is required that you complete an experiment, not an observational study. Explain why your study will be an experiment and not an observational study, as defined on Notes 1.
   * My study will be an experiment and not an observational study because I would be assigning the conditions.

Anticipating sources of variation, bias, and confounding

1. [2] Describe your process of collecting a few preliminary response values. If this was not possible, explain what you did instead. Did this process help you consider any issues in carrying out the experiment you did not consider before?   
     
   While I was testing I wondered whether I should keep the fans on my constant setting (max speed) while changing the voltage and Clock Speed between tests. I decided I should leave the fan speed at the constant (max speed) at all times during testing in order to help get rid of any temperature variations between tests when I am changing the clock speed and CPU voltage.
2. [3] Describe your measurement process (i.e. of obtaining the response values), including what steps you took to minimize measurement error.  
     
   The synthetic program for measuring performance of the CPU, Cinebench R23, will give me a performance score (like 6800) at the end of each test run and as long as I set the clock speed and voltage offset correctly there should no chance of a measurement error.
3. [1] To what degree are the units representative of the question you seek to answer?

While the CPU and the laptop I am using may have similar or exact components to other laptops with the same name and specifications, there is a thing in the computer industry called ‘Silicon lottery’ where the performance of each CPU and therefore computer can differ because the materials that make each CPU will be slightly different. This might allow the same CPU in one computer to achieve higher scores than the same CPU in another computer. So, in theory my testing with my CPU should be representative with all other CPUs of the same spec but unfortunately due to the ‘Silicon lottery’ that’s not really the case.

1. [2] To what degree are the units uniform (to limit variability)? What did you do to make them more uniform?  
    Since I’ll be using my laptop for these runs I’ll keep the clock speeds and the voltage of the CPU the same while making sure fan speeds do not change from run to run as well as having the computer sit on my desk every time I get test results so that my desk surface should heat up at the same pace every time. And I will be measuring the ambient temperature to make sure its roughly the same from run to run.
2. [2] Did blocking of units make sense for your experiment? Explain why or why not. If so, how did you block?  
    No, blocking of units does not make sense for my experiment since each run of Cinebench R23 will be exactly the same for each of the tests.
3. [2] What other nuisance factors are present? (factors other than those of interest that might cause variation/bias in the response) What did you do about them?   
     
   One other Nuisance factor that might be present is the ambient temperature surrounding my computer. Since It’s known that excessive heat is bad for a computer and can cause it to run slower it’s imperative that I keep the ambient temperature around my computer the same for all tests. In addition, since I’ll be testing my laptop on my desk It’s important for my desk to be the same temperature at the start of each test run. So, to make sure these factors don’t influence my results I am going to be keeping a thermostat on my desk to make sure the ambient temp is consistent. I will also be doing several test runs in order to heat up my computer and the desk it’s sitting on to keep the temperature as consistent as possible throughout all the trials otherwise the later results might be disproportionately lower since the computer and desk would have been allowed to heat up.
4. [1] Identify at least one nuisance influence that you recorded as part of your dataset.

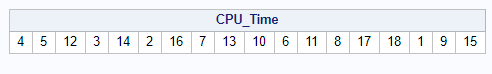
One nuisance influence that I am going to record as part of my dataset is going to be trial number to make sure the responses are roughly the same regardless of when test run was conducted.

Random assignment / the Design

1. [1] How many treatments did you have? How many replications (units) per treatment? How many total observations?

|  |  |  |  |
| --- | --- | --- | --- |
| Offset Voltage (Mv) of CPU | CPU Clock Speed (GHz) | |  |
| 3.3 | 3.4 | 3.5 |
| 0 (essentially default voltage) | 0, 3.3 | 0, 3.4 | 0, 3.5 |
| -80.1 | -80.1, 3.3 | -80.1, 3.4 | -80.1, 3.5 |
| -160.2 | -160.2, 3.3 | -160.2, 3.4 | -160.2,13.5 |

Based on this diagram I will have 9 different treatments (3 for CPU and 3 for voltage) doing 2 replications per treatment would leave me with a total of 18 observations

1. [1] What is the design name? Your options are:
   * **two-way completely randomized (CR[2]) design – if no blocking**
   * two-way complete block (CB[2]) design – if one blocking factor
   * Other (you probably won’t use these):
     + Two-way Latin square (LS[2]) design – if two blocking factors
     + Split plot / Repeated measures (SP/RM) design – if units of two different sizes
2. [2] Show output from PROC PLAN that you used in randomly assigning treatments to units. What was the seed that SAS used (not supplied by you)?  
     
   

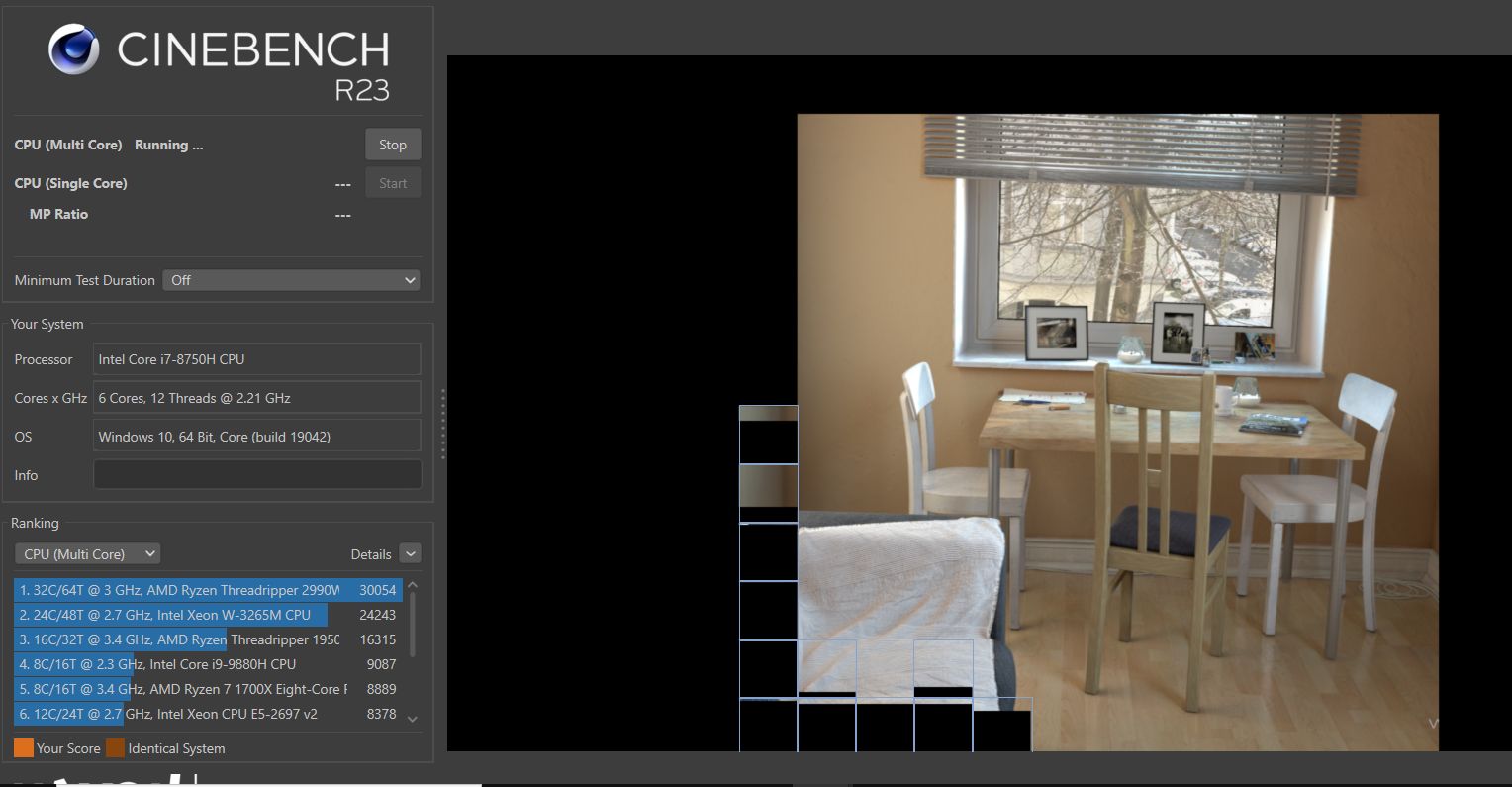
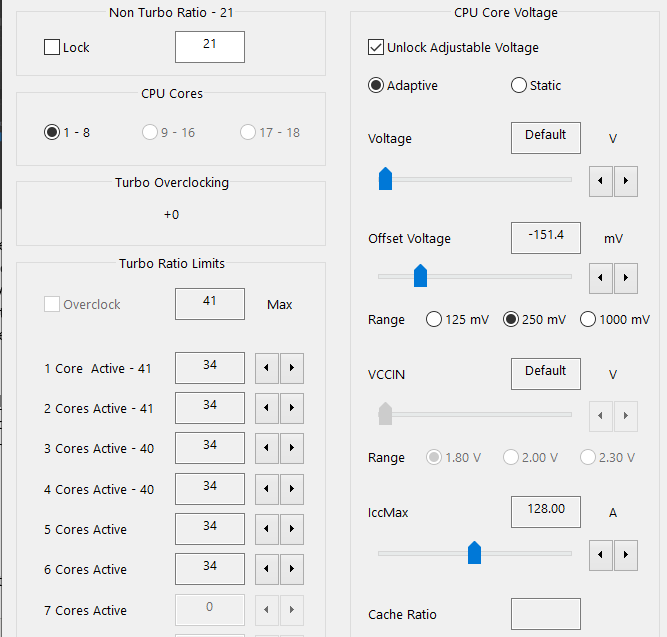
Seed = 164351913

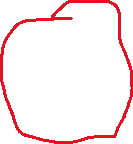
1. [2] Explain how you used this PROC PLAN output to randomly assign treatments to units.  
     
   Okay so using this chart I marked off each treatment with 2 numbers. And depending on where those numbers come up in the sequence in the chart above I will be testing that combination. Example, since the first number is 4 I will test the CPU performance by using a voltage at 0 and CPU clock speed at 3.4

|  |  |  |
| --- | --- | --- |
| 0, 3.3 (1-2) | 0, 3.4 (3-4) | 0, 3.5 (5-6) |
| -80.1, 3.3 (7-8) | -80.1, 3.4 (9-10) | -80.1, 3.5 (11-12) |
| -160.2, 3.3 (13-14) | -160.2, 3.4 (15-16) | -160.2,13.5 (17-18) |

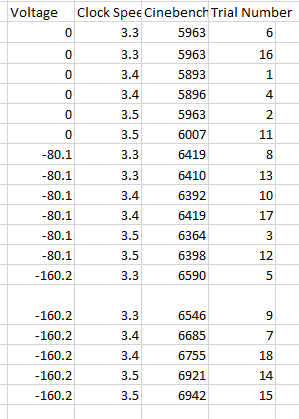
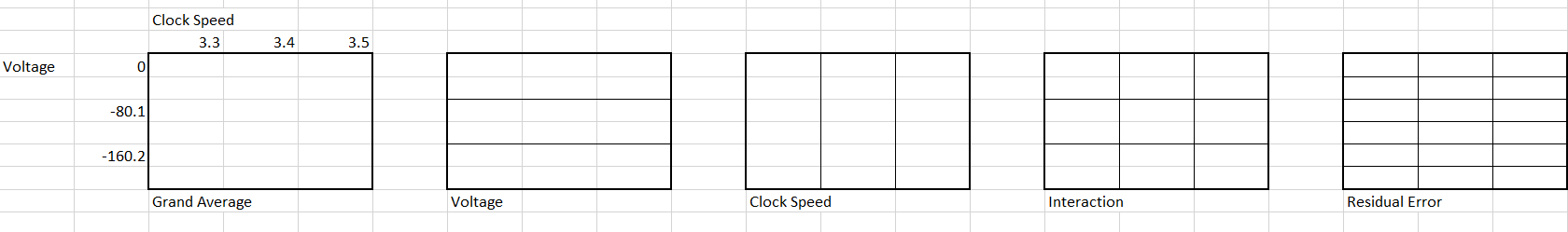
Data collection

1. [4] Describe your process of data collection, providing enough detail that someone else could repeat your experiment.  
     
    First, in order to make sure the computer and table is saturated with heat **I ran “Cinebench R23” 5 times** on normal fan settings at an uncapped GHz and default voltage before recording any results. After that I’ll be using a piece of software called throttlestop to artificially set my CPU clock speeds at the desired level (as 3.3, 3.4 GHz ect.) as well as to lower the CPU Offset voltage by moving the slider to the left. Then I’ll launch the application called “Cinebench R23” (using it to measure CPU performance) and set the program to run only once, click start next to the ‘CPU (Multi Core)’ button and wait for the benchmark to finish and record the score.

[2] Include at least two photos here that help to describe your experiment, writing about what they show.  
  
   
  
The picture on the left shows the synthetic benchmark ‘Cinebench R23’ that I will be using to measure the performance of my intel CPU and you can see the picture with the table and chairs is the imagine that my CPU is trying to render in and once that picture is complete Cinebench R23 will give my CPU a score. On the bottom left you can see the scores that other CPUs have scored on this same benchmark. The picture on the right is “ThrottleStop” and in the red circle you can see where I can adjust the clock speed for my CPU. In the green circle you can see the voltage offset where I move the slider to the left giving me a negative offset which lowers the amount of power the CPU gets which, in theory, should lower the CPU temperature that’ll also give me better performance scores.

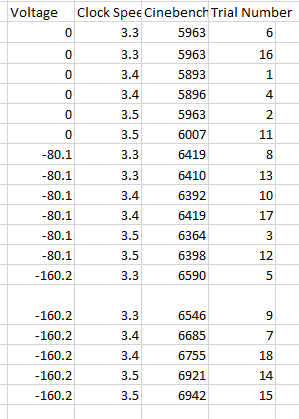


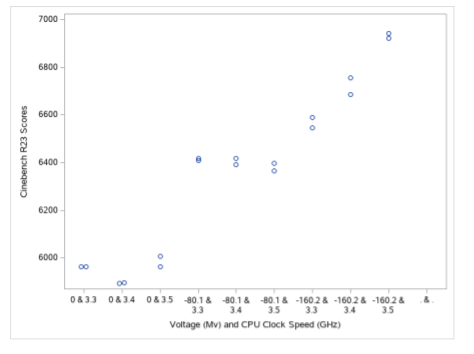
Structure of the data analysis

1. [2] Draw a table (with filled in response values) for the data you got from your experiment. Each cell will correspond to a unit in the experiment; you should label the data table so it is clear which levels of the factors correspond to each cell.   
     
   
2. [2] Based on this data table, draw the factor diagram for your design as in Assignment 
3. [2] What are the degrees of freedom associated with each factor? You are required to have at least 6 degrees of freedom for residual error.

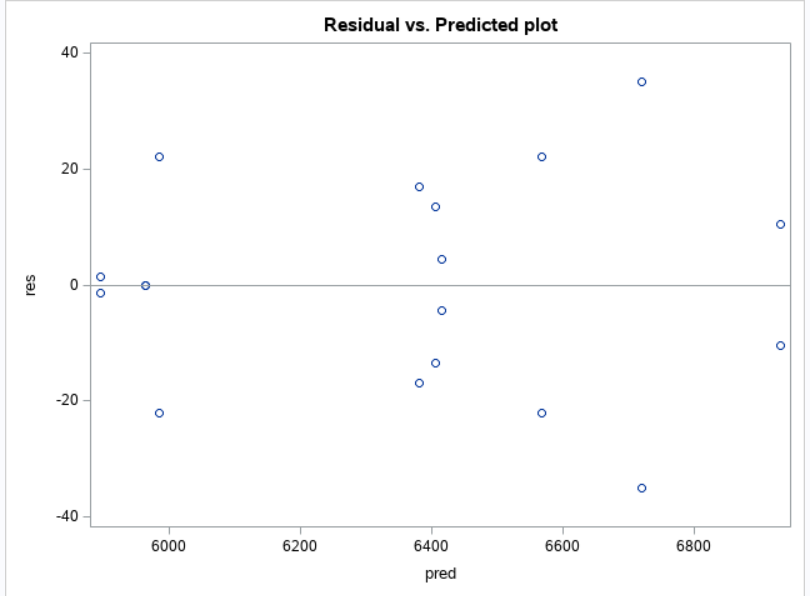
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Factors | Outside factors | # Levels | Sum of df for outside factors | df |
| Grand avg | None | 1 | 0 | 1 |
| Clock Speed | Grand Average | 3 | 1 | 2 |
| Voltage | Grand average | 3 | 1 | 2 |
| Interaction | Voltage, Clock Speed, Grand average | 9 | 5 | 4 |
| Residual Error | Grand average, interaction, voltage, clock speed | 18 | 9 | 9 |

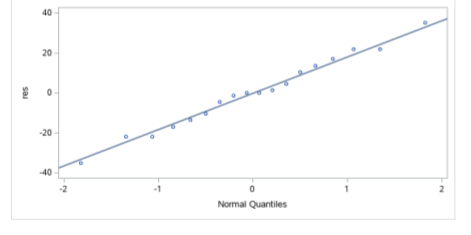
Data analysis and interpretation

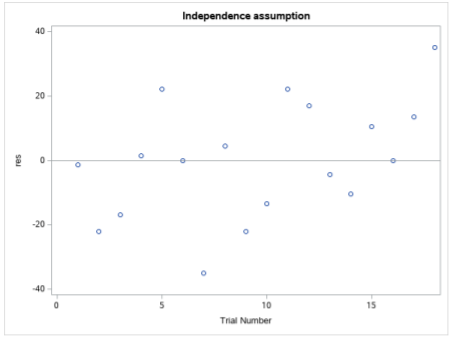
1. [2] Copy and paste the Excel file that shows the dataset used to analyze the data in SAS. This should include variables for each factor, the response, the nuisance influence(s) you recorded, and the blocks (if applicable).  
   
2. [2] Copy and paste a parallel dot plot of the dataset. The x-axis should have locations for each of the treatments and the y-axis should show the response. Use formats and labels so the variables and treatments are described clearly (and are not shown as abbreviations). If blocking is part of your data, this should be a connected parallel dot plot.



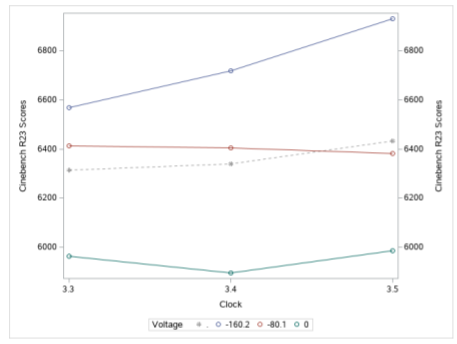
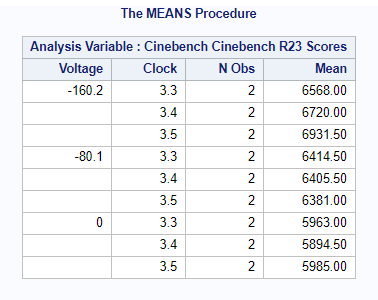
1. [2] Interpret this parallel dot plot in terms of the means and spreads of the responses in each of the treatments.  
    Mean response of Cinebench R23 score tends to increase going left to right as the voltage goes down and once the voltage hits -160.2 Cinebench scores increase again as the clock speed increases. The spread of the responses is fairly similar for voltages of 0 and -80.1 Mv regardless of clockspeed, however, once voltages hit -160.2 there is a lot of spread as the CPU clock speed increases.
2. Check the assumptions. Copy and paste each plot mentioned below and state why it gives evidence the given assumption is satisfied or violated.
   * [1] Constant standard deviation and the residual vs. predicted plot.

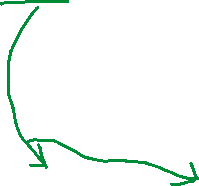
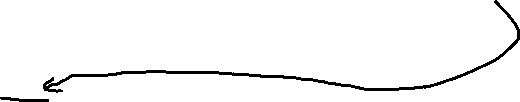
  
  
The constant standard deviation assumption is not met because the spread of the residuals is not constant as there is less spread on the outsides of the graph than in middle of the graph.

[1] Normality assumption and normal quantile plot  
   
On the normal quantile plot, the points follow the line extremely well so the normality assumption is met.

* + [1] Independence assumption and plot of residuals vs. your nuisance influence (such as trial number).   
    

I do not see any patterns based off of this graph, which means that the independence assumption is thoroughly met as trial number had no effect on the outcome of the data.

1. Interaction
   * [1] Copy and paste an interaction plot made by SAS.   
       
     
   * [1] Copy and paste a table of the means of the response for each treatment combination.  
     
2. [3] Explain whether or not interaction is present in your dataset in terms of **both** the interaction plot and the table of means above.  
    An interaction is present in the interaction plot since no where in the graph are the lines entirely parallel although they get close when the voltage is at 0 and -80.1 going from clock speeds 3.3 to 3.4 GHz. We can more clearly see that an interaction is taking place by looking the table of means I converted into this nice chart below and subtracted horizontally and vertically and compared those differences together to see that they were all different values indicating an interaction.

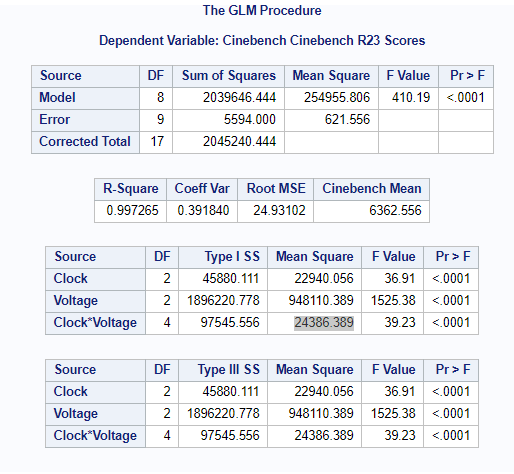


|  |  |  |  |
| --- | --- | --- | --- |
| Offset Voltage (Mv) of CPU | CPU Clock Speed (GHz) | |  |
| 3.3 | 3.4 | 3.5 |
| 0 (essentially default voltage) | 5963.00 | 5894.50 | 5985.00 |
| -80.1 | 6414.50 | 6405.50 | 6381.00 |
| -160.2 | 6568.00 | 6720.00 | 6931.50 |

voltage 0 = -5,916.5 Clock 3.3 = -7,019.5

voltage -80.1 = -6,372 Clock 3.4 = -7,231

voltage -160.2 = -7,083.5 Clock 3.5 = -7,327.5

1. [2] Copy and paste SAS output that gives as much of entries in the ANOVA table as possible.  
   
2. [3] Fill in the ANOVA table below for your dataset. If your design does not include blocking, delete that row.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Source | df | SS | MS | F-ratio | P-value |
| Grand avg | 1 | X | X | X | X |
| Clock | 2 | 45880.11 | 22940.06 | 36.91 | <.0001 |
| Voltage | 2 | 1896220.78 | 948110.39 | 1525.38 | <.0001 |
| Interaction | 4 | 97545.56 | 24386.39 | 39.23 | <.0001 |
| Residual | 9 | 5594 | 621.56 | X | X |
| Total | 18 | x | X | X | X |

1. [2] Interpret the F-ratios for each factor and for interaction.  
     
   The average variation due to GPU Clock Speed is 36.91 times the average variation due to residual error.

The average variation due to voltage is 1525.38 times the average variation due to residual error.

The average variation due to the interaction of voltage and CPU Clock Speed is 39.23 times the average variation due to residual error.

1. [2] Find the residual standard deviation and interpret it within the context of the experiment.  
     
   Residual standard deviation = 24.93 On average, the distance between the observed and predicted CineBench R23 Score for the CPU is 24.93 points
2. [1] If you used blocking, evaluate how effective it was in reducing the unexplained variation – i.e. the sum of squares due to residual error. (You did not use blocking, just write “not applicable”.)

“not applicable”

1. [2] Should you interpret the main effects or simple effects of the factors of interest? (This decision should be based on a hypothesis test at the 0.05 significance level.) Justify your decision.  
     
   I should interpret the simple effects of the factors of interests since the p value for the interaction effect is <.0001 which is less than our alpha of .05
2. [4] Regardless of your answer above, find point estimates of … (Hint: these are all differences in sample means; be sure to state which levels are greater/less)
   * the main effects of each factor



|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | CPU Clock Speed (GHz) | |  |  |  | CPU Clock Speed (GHz) | |  |
| Offset Voltage | 3.3 | 3.4 | 3.5 |  | Offset Voltage | 3.3 | 3.4 | 3.5 |
| 0 |  | 5947.5 |  |  | 0 |  |  |  |
| -80.1 |  | 6400.333 |  |  | -80.1 | 6315.167 | 6340 | 6432.5 |
| -160.2 |  | 6739.833 |  |  | -160.2 |  |  |  |

Main effect Voltage = 5947.5 - 6400.333 - 6739.833 = **-7192.67**(voltage -160.2mV is the greater than 0 & -80.1 mV)

Main effect for Clock Speed = 6315.167 – 6340 - 6432.5 = **-6457.33** (Clock speed 3.5GHz is the greater than 3.3 & 3.4 GHz)

* + **the simple effects of each factor**

Simple effect for Clock speed when:

voltage 0 = -5,916.5 (3.5 is greater than 3.3 & 3.4)

voltage -80.1 = -6,372 (3.3 is greater than 3.5 & 3.4)

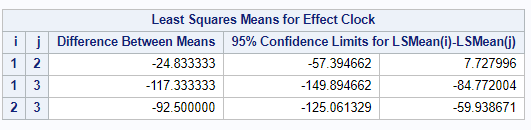
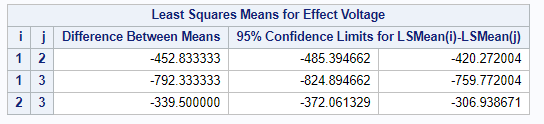
voltage -160.2 = -7,083.5 (3.5 is greater than 3.3 & 3.4)

Simple effect for Voltage when:

Clock 3.3 = -7,019.5 (voltage -160.2 is greater than -80.1 & 0)

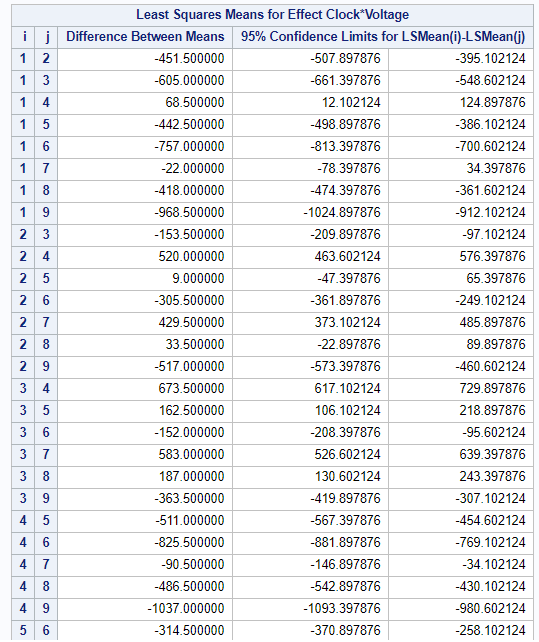
Clock 3.4 = -7,231 (voltage -160.2 is greater than -80.1 & 0)  
 Clock 3.5 = -7,327.5 (voltage -160.2 is greater than -80.1 & 0)

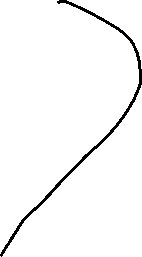
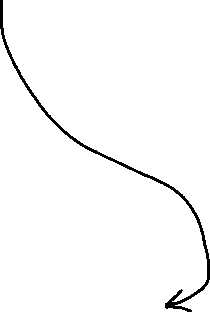
1. [4] Copy and paste 95% confidence intervals from SAS …
   * For the main effects

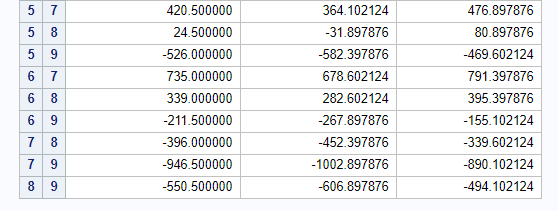
  




* + For the simple effects





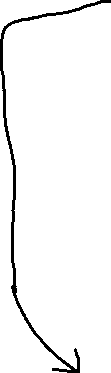
­­



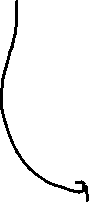
1. Interpretation of confidence intervals.
   * [2] Choose one of these confidence intervals for the **main effects** and interpret it within the context of the problem.

We are 95% confident that the mean CineBench R23 Score with a voltage set at -80.1 mV is between --420.27 and 485.39 points more than the mean CineBench R23 score with the voltage set at its default of 0 mV.

* + [2] Choose one of these confidence intervals for the simple effects and interpret it within the context of the problem.  
      
    Simple effect of Voltage (from 0mV to -160.2mV: 7 to 9) when CPU clock speed is 3.5 GHz is: (890.10, 1002.89). Focusing on when the CPU clock speed was 3.5 GHz, we are 95% confident that the average Cinebench R23 score when voltage is -160.2mV is between 890.10 and 1002.89 points more than when voltage is at 0 mV.



* [2] Choose one of these confidence intervals above and show how it is calculated from the formula.  
   Simple effect:   
  Estimate t\* x SE t\* = 2.2621571628



SE = SD x leverage factor 24.93(sqrt(1/2 +1/2)) = 24.93



946.5+/- 2.2621571628 (24.93) = (890.10, 1002.89)



1. [2] Write a paragraph summarizing your results. Overall, how did your two factors affect the response?  
     
   What I found during my experiment is that both factors (Voltage and clock) significantly affected the CPU performance as measured by the scores from the synthetic benchmark Cinebench R23. This occurred when voltage decreased and clock speed increased. However, it’s important to note that while both factors were significant in affecting the CPU performance, tuning the voltage down affected CPU performance more than changing the clock speed since it’s F-value is significantly larger at 1525.38 compared to the Clock speed’s F-value of 36.91. We can also see evidence of this in the interaction plot posted on problem 26 where clock speed stays relatively consistent (until the voltage hits -160.2 where clock speed starts to have a greater impact on Cinebench scores) while changes in voltage consistently illustrates the starkest difference between CPU performance scores.

Conclusion / What you learned

1. [1] What did you expect the results of your experiment to be and is that what happened?  
     
   I expected that as voltage decreased and the clock speed increased that the Cinebench R23 scores for the CPU would increase. That is mostly what happened but voltage played a much bigger role in increasing the scores of the cinebench R23 scores while clock speed didn’t play much of a factor until the voltage hit -160.2 mV.
2. [1] What problems arose during data collection and what would you have done differently if you could do your experiment again?  
     
   After refining this project from project 1, I had pretty much solved the problems I encountered during data collection last time and no new problems arose during this experiment as everything went surprisingly smooth.
3. [1] Were there unexpected sources of variation, bias, and/or confounding you didn’t anticipate? What were they and what did you do about them?

Nope, not at all. Like I detailed in the previous question all the biases/confounding stuff I had found during project 1 was ironed out and refined so that they would not happen again during project 2 and no new problems arose during this experiment.

1. [1] What further questions/study are inspired by your experiment?  
   One study that I would like to do in the future that was inspired by this experiment is testing ‘real world performance increases’ of changing factors such as the CPU clock speed and voltage to see how it affects a video games performance. So, for project 1 and 2 I tested CPU performance on a synthetic benchmark (Cinebench R23), but in the future I would like to see how these changes directly relate to FPS (Frames per second – which makes the game look smoothers) increases in an actual game.