



# EXPLORING MICROPROCESSOR DESIGN TRENDS WITH UNSUPERVISED MACHINE LEARNING

Jason Glass

NORTHWESTERN EECS MS PROJECT REPORT Advised by Gokhan Memik and Russell Joseph



Table 1: Data Profile.....	5
Table 2: Machine Learning Results .....	6
Table 3: SPEC CINT2006 Decision Table .....	13
Figure 1: Root Mean Squared Error (RSME) .....	2
Figure 2: Normalized Root Squared Error (RMSD).....	3
Figure 3: L2 Cache Size Decision Tree .....	7
Figure 4: L2 Cache Size Correlating Traits .....	7
Figure 5: I/O Gt/s Decision Tree .....	8
Figure 6: I/O Correlating Traits .....	8
Figure 7: Release Date Decision Tree.....	9
Figure 8: L3 Cache Size Decision Tree .....	9
Figure 9: L3 Cache Size Correlating Traits .....	10
Figure 10: Socket Type Decision Tree .....	10
Figure 11: Socket Type Correlating Traits .....	11
Figure 12: Number of Cores Decision Tree .....	11
Figure 13: Number of Cores Correlating Traits .....	12
Figure 14: Price Category Decision Tree .....	12
Figure 15: Price Category Info Gain .....	13

## Background

This paper uses unsupervised machine learning and basic statistics to determine the viability of predicting microprocessor specifications and performance based off a limited number of design attributes and traits.

Unsupervised machine learning is defined by the task of inferring a function that describes unclassified data [1]. Another interpretation is that neither the algorithm nor the programmer (supervisor) is sure of the correct result. Since this project is about the genesis of a predictive algorithm that dichotomy fits the project well. Supervised in this example would imply the programmer already knows a formula to predict computer performance based off the design attributes. In the future this work could be applied to a supervised learning algorithm to achieve more specific results, but that is outside the scope of this paper.

Weka was used for this project to conduct the unsupervised learning, Weka is an open-source and free machine learning software suite that allows for quick and powerful analysis of large datasets through exploratory data mining [2]. The J48, M5p and RepTree algorithms were used for this project through Weka. The j48 algorithm is an extension of the C4.5 decision tree algorithm, it produces trees through the measurement of information entropy. The RepTree algorithm is similar and uses information gain/variance to sort numerical attributes. The M5p model uses linear regression at every leaf of the regression tree [3].

The Root Mean Squared Error and Normalized Root Squared Error statistics were used to interpret the results of the Weka data mining [4].

$$\text{RMSE}(\hat{\theta}) = \sqrt{\text{MSE}(\hat{\theta})} = \sqrt{\text{E}((\hat{\theta} - \theta)^2)}.$$

*Figure 1: Root Mean Squared Error (RSME)*

$$E_i = \sqrt{\frac{\sum_{j=1}^n (P_{(i)} - T_j)^2}{\sum_{j=1}^n (T_j - \bar{T})^2}}$$

*Figure 2: Normalized Root Squared Error (NRSE)*

## Motivation

The purpose of this project is to apply unsupervised machine learning techniques to a large subset of the Intel Microprocessor catalog. The goal of the machine learning is to determine if the design specifications of a microprocessor maybe determined by some subset of their other traits. For example, can the size of the L2 cache be correctly predicted based on the L3, I/O Bus, Release Date and Price?

The inspiration for this research was originally to explore for exponential trends in computing, such as Moore's Law, with an emphasis on generating a machine learned algorithm to predict performance on CINT2006 based purely on design specifications.

The value of this research is two-fold, the ability to predict microprocessor design traits purely predicated on its other traits provides valuable insight into the variability (or lack thereof) of different niche of processors. The second point of value is the potential for an all-encompassing metric that can predict processor performance without having to simulate anything. This would not supersede the necessity for testing and simulation but would provide an interesting way to design a microprocessor based on design specs. This could be of immense value in the design of future heterogenous chip systems. This work was motivated by research that correlated cache redundancy design with profit [5].

## Methodology

Microprocessor design specifications were pulled from the Wikipedia Intel pages for the Xeon processors, 193 processors were used in the following data set. The following 12 traits were used:

1. Release Date
2. Cores
3. Threads/Core
4. Frequency
5. TDP
6. Price Category (Price Normalized to 1000\$ increments)
7. L2 Cache size
8. L3 Cache Size
9. Socket Type
10. I/O Bandwidth
11. Memory Bandwidth
12. CINT2006 Result

Weka 3.8 was used for the machine learning classifier algorithms. The RepTree was used for Numeric Attributes and J48 Tree used for Nominal attributes for the first 11 listed traits used. A decision table was used for predicting SPEC CINT2006 performance after a brute force testing pattern to determine the best classifier.

Each classifier was run trying to predict one attribute based off the other 10 given, this was to find the attributes with the highest information and correlation value. Each classifier was run with 5 random seeds and 10-fold cross validation and the best results were picked for each. Data such as price category was normalized into 1000\$ increments to enhance the classifiers accuracy and provide more nominal attributes, which tend to generate more accurate predictions in Weka.

The final step was then the brute force testing of a classifier to predict performance, a simple two-factor decision tree produced the highest correlation.

## Results

*Table 1: Data Profile*

Attribute	Min	Max
Release Date	3/6/2012	6/1/2016
Cores	2	22
Threads/Core	1	2
Frequency (Ghz)	1.6	3.7
TDP (W)	5	165
Price Category	0	7
L2 Cache (Kib)	512	4608
L3 Cache (MiB)	5	45
Socket Type	LGA 2011, LGA 2011-1, LGA 2011-3, LGA 1356	
I/O (Gt/s)	12.8	64
Memory (MB/s)	3999	9600
CINT2006	33.4	75.1

Table 2: Machine Learning Results

Trait Predicted	Best Correlation/Accuracy	Normalized Root Square Error	Std. Dev of Error	Variance	Classifier Type
L2 Cache	0.9681	21.5689	1.8398	6.769728	m5p
I/O	0.9715	21.9686	0.79785	1.273129	m5p
Release Date	0.9731	21.9881	0.45235	0.409241	REPTree
L3 Cache	0.9425	32.9992	0.4069	0.331135	m5p
Cores	0.8919	40.5847	2.48155	12.31618	REPTree
Price Category	0.8757	47.7504	0.45235	0.409241	REPTree
Frequency	0.8443	51.9323	0.87485	1.530725	Reptree
TDP	0.8385	54.1985	0.02785	0.001551	m5p
Memory	0.8271	55.9914	0.0437	0.003819	m5p
Socket Type (acc)	0.8497	63.7375	1.8149	6.587724	j48
CINT2006_table	0.6330	75.1222	0.839434626	0.880813	Decision Table
Threads/Core	0.4377	77.5449	3.859664468	18.62126	m5p
CINT2006_m5p	0.4393	88.5414	0.793538153	0.787128	m5p

The following figures are a subset of the results that were chosen for accuracy and ease of interpretation:

Correlation = 96.81%, RSME = 21.57%

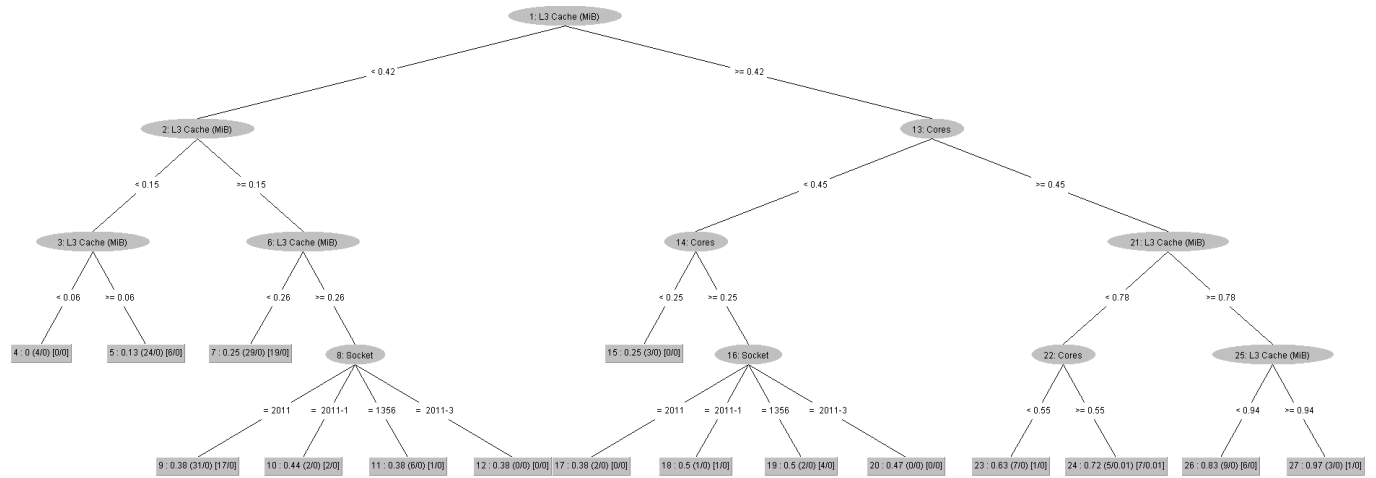


Figure 3: L2 Cache Size Decision Tree

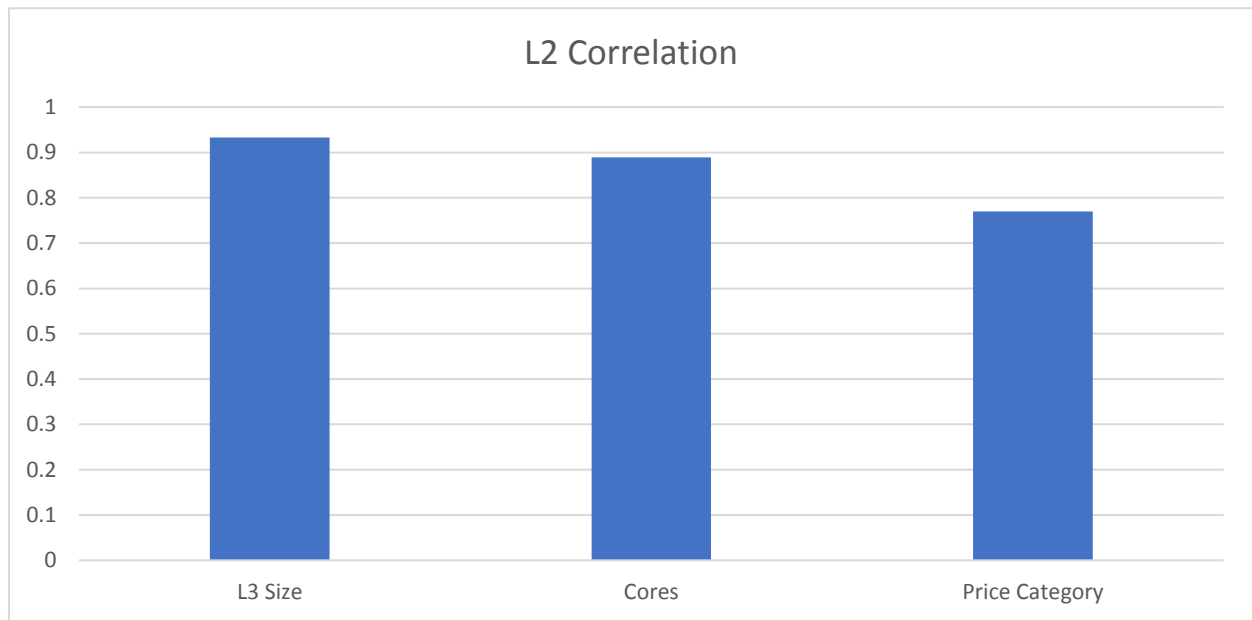


Figure 4: L2 Cache Size Correlating Traits



I/O, Correlation =97.15%, RSME = 21.97%

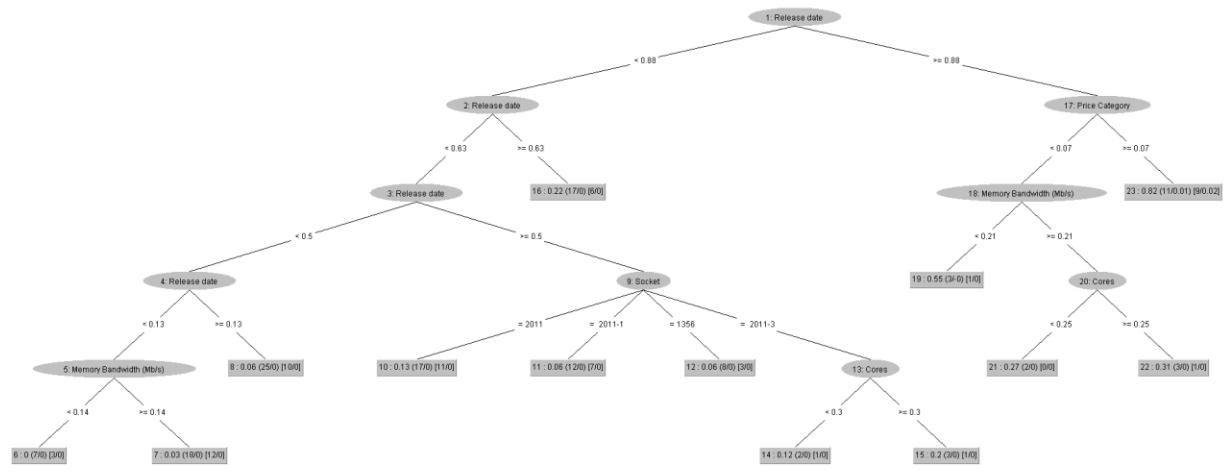


Figure 5: I/O Gt/s Decision Tree

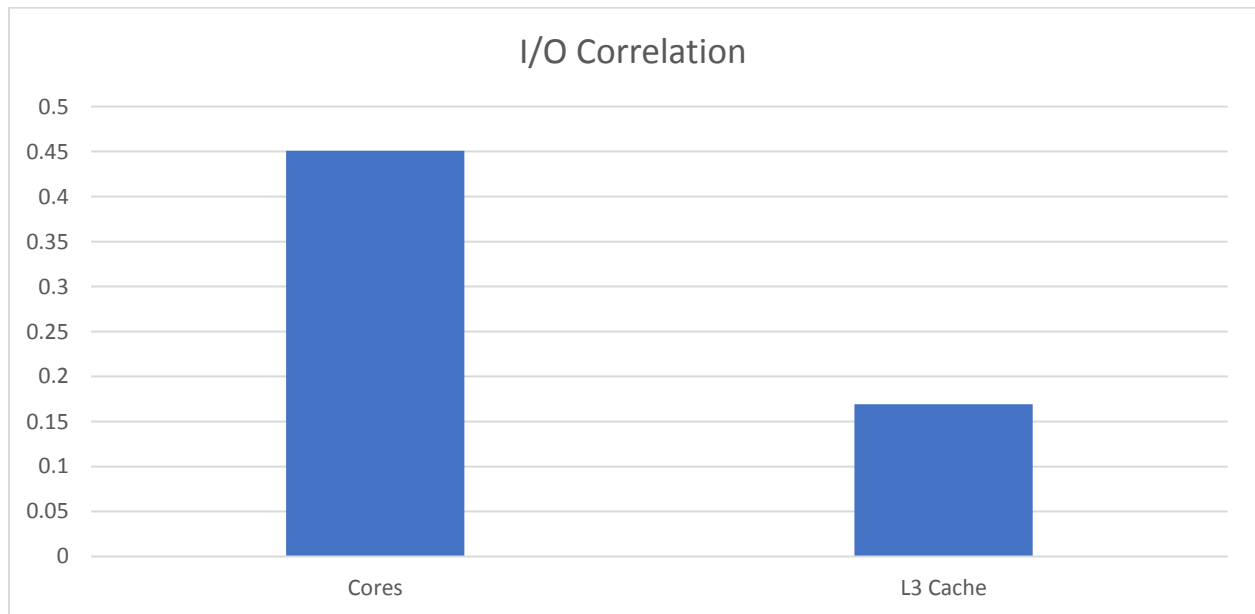


Figure 6: I/O Correlating Traits

Correlation = 97.31%, RSME = 21.98%

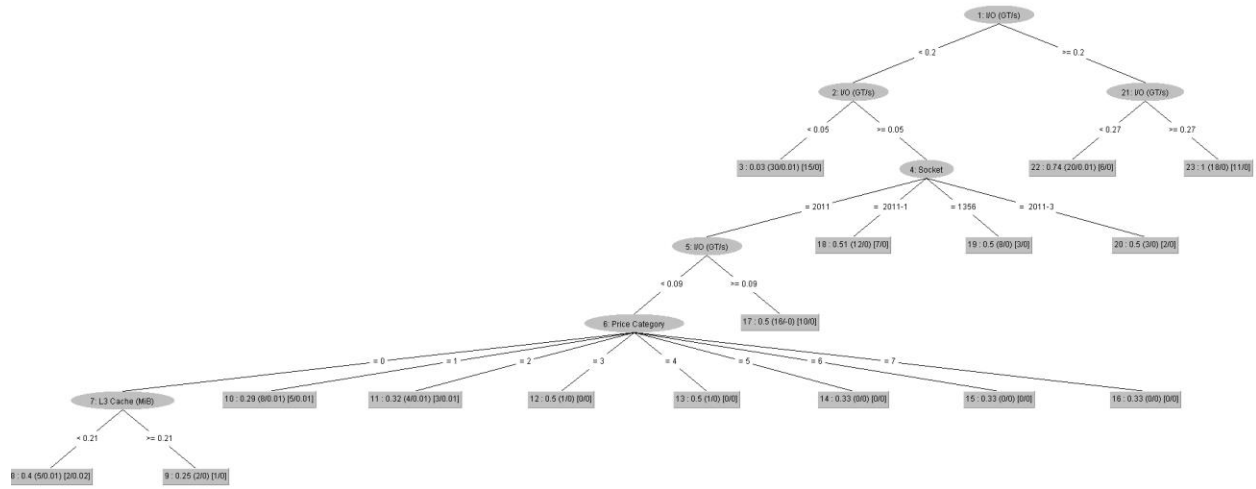


Figure 7: Release Date Decision Tree

Correlation = 94.25%, RSME = 33.00%

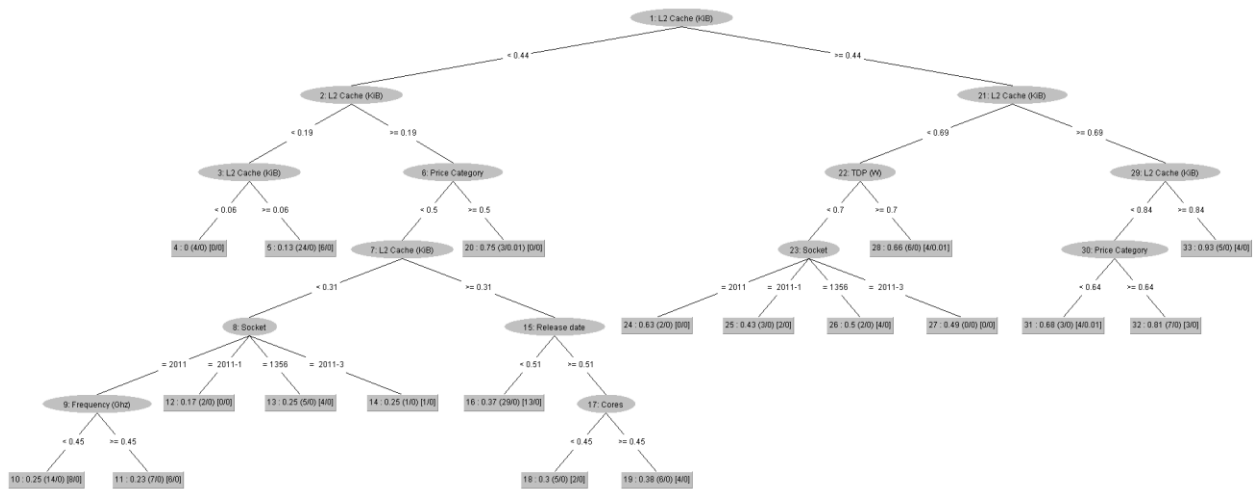


Figure 8: L3 Cache Size Decision Tree



Figure 9: L3 Cache Size Correlating Traits

Socket Type, Acc = 84.97%, NRSE = 67.3673%

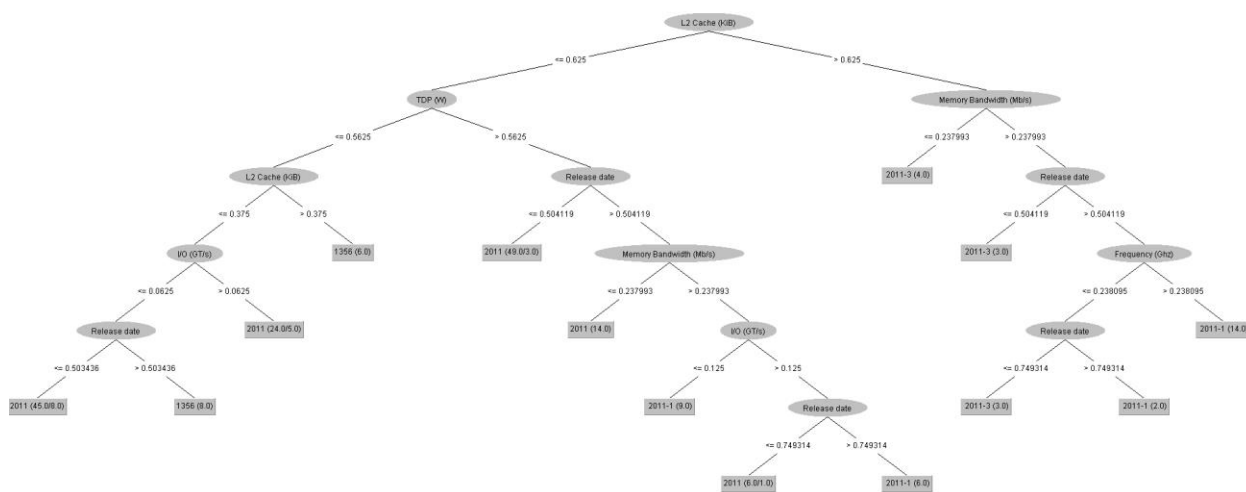


Figure 10: Socket Type Decision Tree

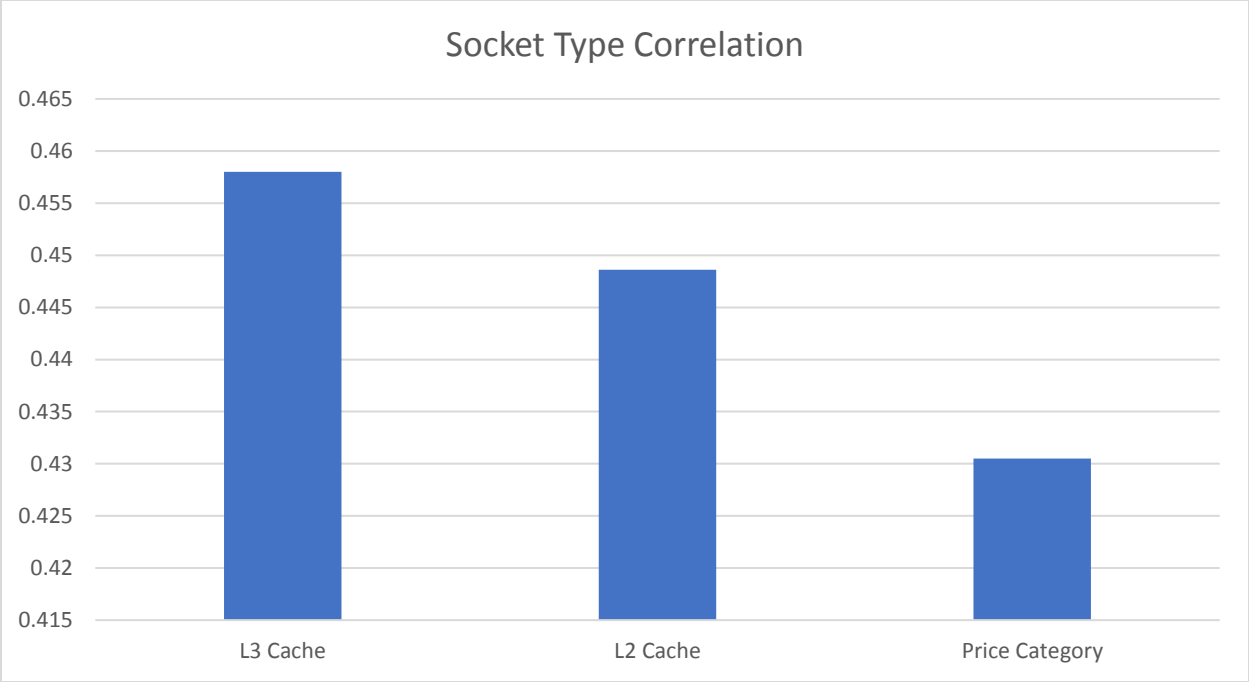


Figure 11: Socket Type Correlating Traits

Correlation = 89.18%, NRSE = 39.65%

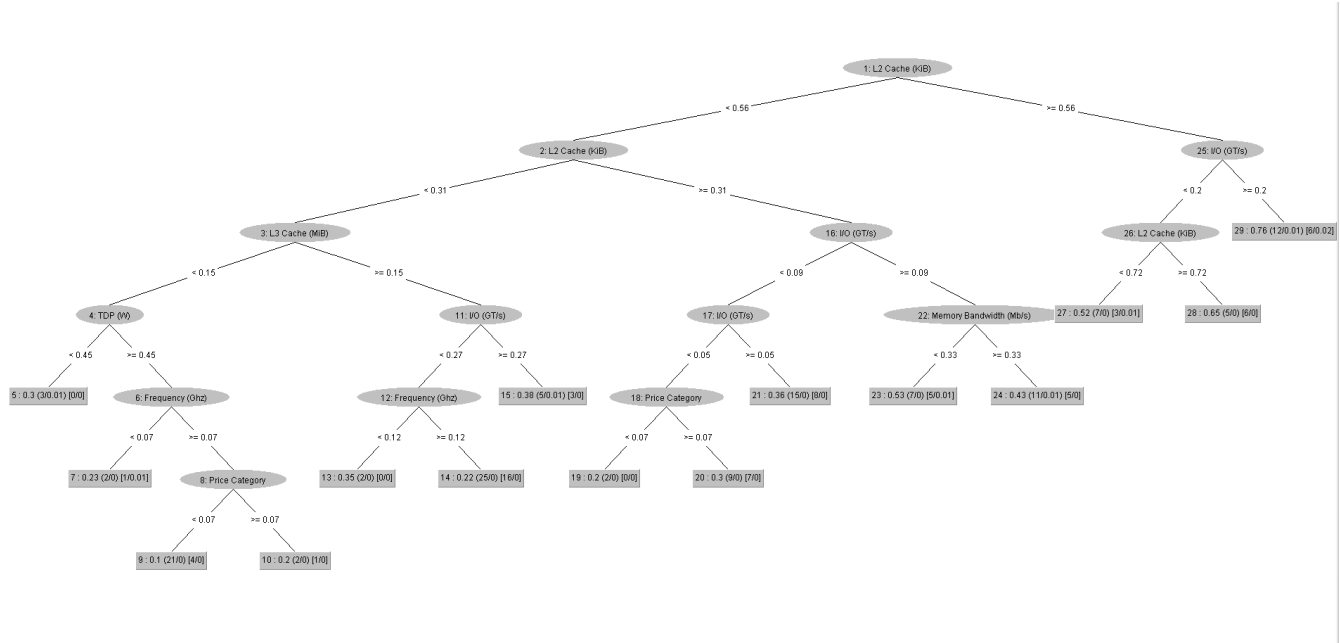


Figure 12: Number of Cores Decision Tree

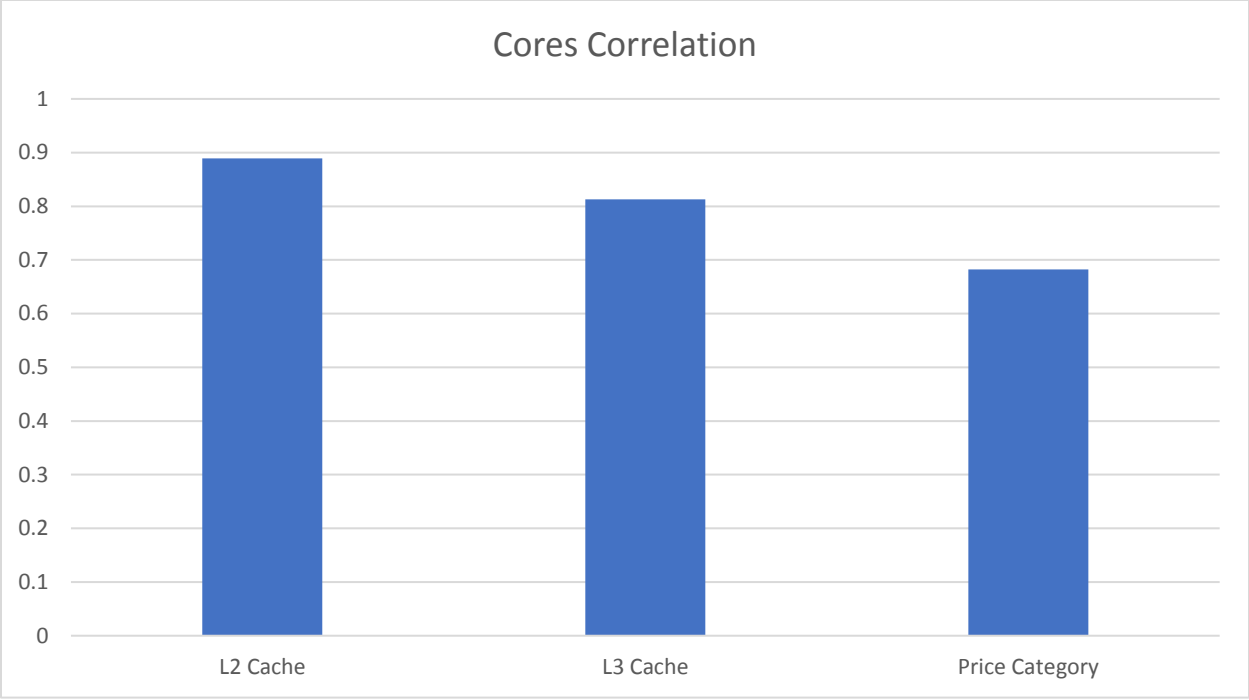


Figure 13: Number of Cores Correlating Traits

Price Category, Correlation = 87.57%, NRSE = 45.91%

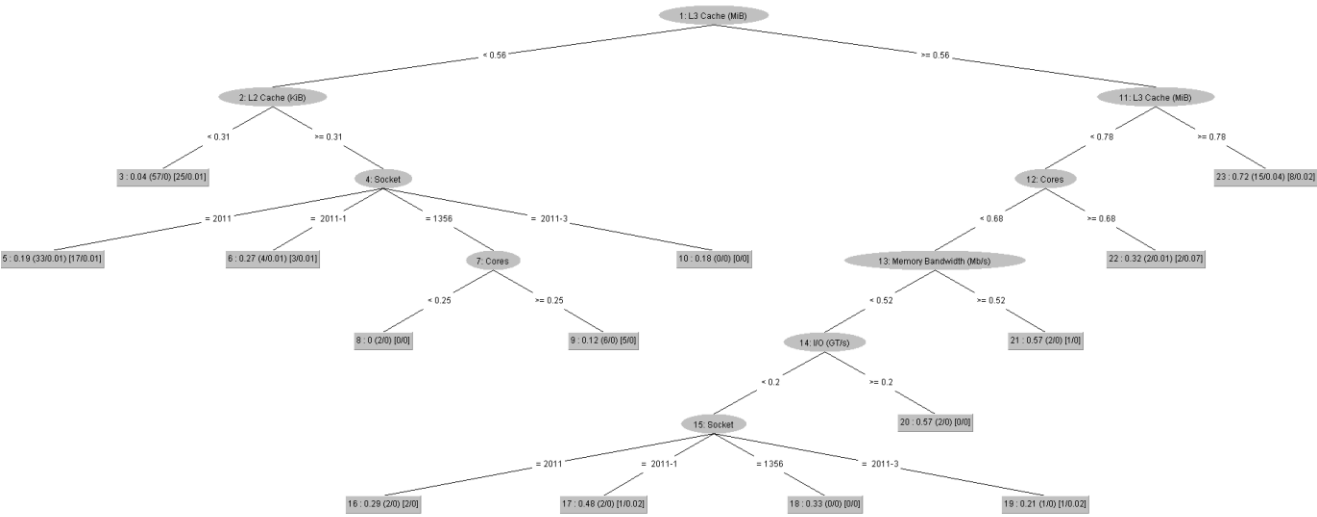


Figure 14: Price Category Decision Tree

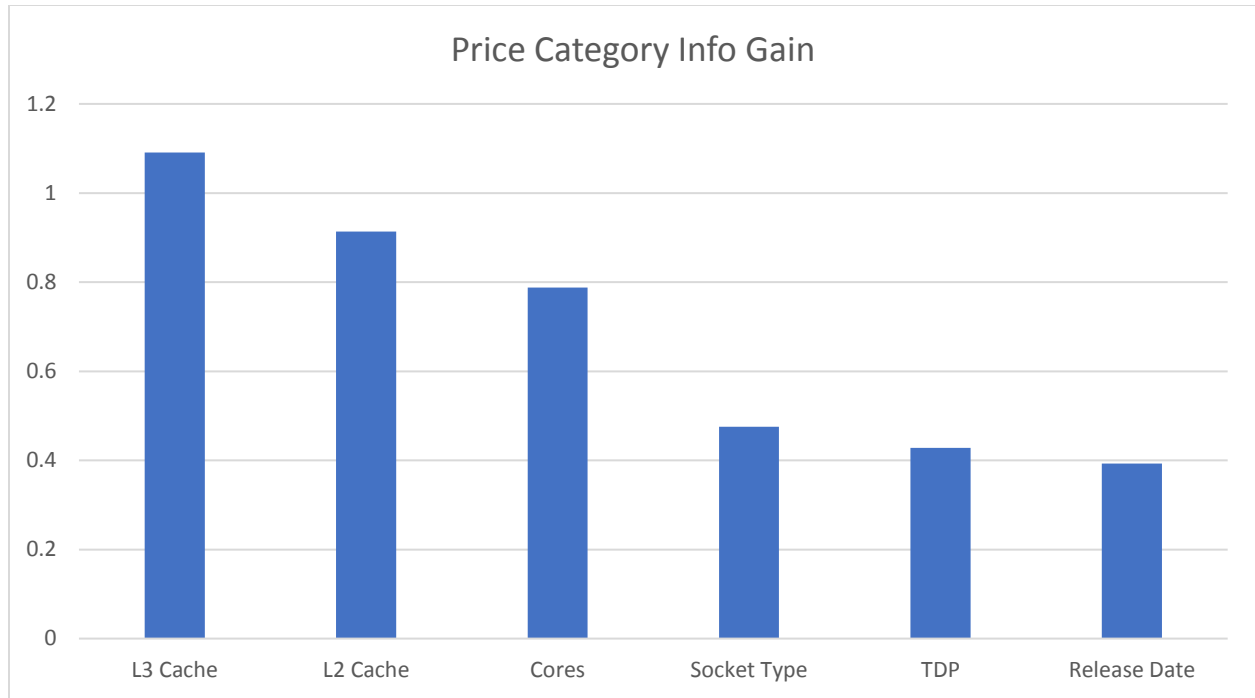


Figure 15: Price Category Info Gain

Correlation =64.77%, RSME = 76.15%

Table 3: SPEC CINT2006 Decision Table

I/O (GT/s) range	Memory Bandwidth (Mb/s)	CINT2006
'(0.1-0.2]'	'(0.9-inf)'	74.26666666666667
'(-inf-0.1]'	'(0.8-0.9]'	47.9
'(0.1-0.2]'	'(0.8-0.9]'	60.01999999999999
'(0.2-0.3]'	'(0.6-0.7]'	58.5
'(0.3-0.4]'	'(0.6-0.7]'	74.8
'(-inf-0.1]'	'(0.6-0.7]'	54.74285714285714
'(0.3-0.4]'	'(0.4-0.5]'	65.4
'(0.2-0.3]'	'(0.4-0.5]'	54.45200000000000
'(0.1-0.2]'	'(0.4-0.5]'	12.18000000000000
'(-inf-0.1]'	'(0.4-0.5]'	56.13333333333332
'(0.1-0.2]'	'(0.2-0.3]'	43.5
'(-inf-0.1]'	'(0.2-0.3]'	51.97333333333333
'(0.9-inf)'	'(0.1-0.2]'	50.43333333333334
'(-inf-0.1]'	'(0.1-0.2]'	66.5
'(0.7-0.8]'	'(0.1-0.2]'	41.06153846153847
'(0.5-0.6]'	'(-inf-0.1]'	35.28
'(-inf-0.1]'	'(-inf-0.1]'	38.3

## Conclusion

The results of this project were very promising for what could be called “scratching the surface” of exploratory data mining on this subject. The main conclusions to be drawn are that the L2 and L3 cache sizes, Release price and number of cores all covary strongly. The L2 and L3 provide almost 2 bits worth of information towards determining towards price alone.

The L2, L3, release date, I/O speed, number of cores, socket type and price category were all capable of being correlated with the 11 used traits above 85% and for Release Date, I/O and L2 above 96%. These are very high correlations and indicate that these design traits are in fact chosen in conjunction with on another, as would be reasonably expected.

Another interesting conclusion was that given the 11 traits used in the paper, release date can be predicted by the machine learning algorithm within 20% accuracy of the actual date. This is a surprising amount of accuracy, this is a positive indicator towards consistent exponential/linear trends being in the background of microprocessor design and manufacture. While the existence of exponential trends like Moore’s Law is given, it is not an established fact as of now that all computing exists within a well-defined exponential/linear trend.

The last conclusion, based on the final decision table for the SPEC CINT2006 results, is that the memory bandwidth determines the largest portion of the SPEC performance. While the memory bandwidth does strongly correlate with the number of cores, it was surprising that Frequency was not a main factor in the performance, which would’ve been a perfectly valid naïve conclusion based on a fundamental understanding of computing performance.

## Future Work

The clear limitations to this work are scope and completely unstructured unsupervised machine learning. Extensions would be to include the i3, i5, i7 and i9 processor catalogs and to extend the scope of the data set from a current 4-year span to over 20 years. By including more distinct sets of chips it would be possible to perform data clustering and to compare the consistency of design trends across several microprocessor niches.

A further extension would be to utilize more powerful algorithms such as a multilayer perceptron and logistic regression on a more refined nominal data set. However, this classification of nominal attributes is a direct extension of the work of this paper. It can already be concluded from this paper that a metric combining release price, L2 and L3 cache size would provide a large amount of informational gain based off the results of the machine learning.

Lastly it would be prudent to focus more research into the development of an equation or decision tree model to more accurately predict the SPEC CINT2006 performance, which ultimately proved out of reach for this paper with a final correlation of only 64.77% based of the most optimal decision tree model.

## References

- [1] "Supervised and Unsupervised Machine Learning Algorithms." *Machine Learning Mastery*, 22 Sept. 2016, [machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/](http://machinelearningmastery.com/supervised-and-unsupervised-machine-learning-algorithms/).
- [2] Weka 3: Data Mining Software in Java." *Machine Learning Project at the University of Waikato in New Zealand*, [www.cs.waikato.ac.nz/~ml/weka/](http://www.cs.waikato.ac.nz/~ml/weka/).
- [3] "C4.5 Algorithm." *Wikipedia*, Wikimedia Foundation, 10 June 2018, [en.wikipedia.org/wiki/C4.5\\_algorithm](https://en.wikipedia.org/wiki/C4.5_algorithm).
- [4] "Root Mean Square Deviation" *Wikipedia*, Wikimedia Foundation, 10 June 2018, [https://en.wikipedia.org/wiki/Root-mean-square\\_deviation](https://en.wikipedia.org/wiki/Root-mean-square_deviation)
- [5] Das, Abhishek, et al. "Evaluating the Effects of Cache Redundancy on Profit." *2008 41st IEEE/ACM International Symposium on Microarchitecture*, 2008, doi:10.1109/micro.2008.4771807.