Persona in Intel PC System

Abstract

In this project, our goal is to find the relationship between the Persona and their PC system. We are trying to compare the performances of different models and use the features in the system to predict the type of a user. To achieve this, we collect data from the user end, clean the data, explore the data using hypothesis tests and fit our data into some classification machine learning models, and also test the performance and optimizing parameters of our models.

Introduction

With the development of hardware products in the PC market, there are more and more choices for people to choose their PC configuration. For different users, they have different needs for their computers. For example, the gamers need more power on the CPU and graphic cards. For other people like pro video editors or engineers, they may need more memory for them to run some professional applications like Final Cut Pro, CAD, etc.. In our project, our goal is to find the relation between the users' system and the type of the users. Also, we are trying to use the data of the user's system to predict the type of the user.

Background

There are mainly 4 steps: data collecting, cleaning, EDA and modeling. For collecting the data, we use the ATLSDK and XLSDK provided by Intel. In the data we collected, there are some missing and suspicious values in the data. After cleaning, we proceed to EDA to explore the data. We also did some hypothesis testing in some of the features. See the description of all features in appendix 1.1.

Methods

1.Data collection:

To get the data from the PC system, we use the XLDSK and ATLSDK provided by Intel and build an input library. We are given the basic template of data collection by Intel. We added some useful metrics and used some APIs to get the data we want. We mainly collected the data

related to the user's battery, like capacity, percentage remaining, etc.. Intel also provided us the users' system data so that we can have more information about the users.

ATLSDK:

Analyzer Task Libraries (ATLs) are used by the Intel® System Usage Report (SUR) field data collector to expand and customize basic features and collection capabilities. These libraries are the preferred mechanism to supplement the Intel® SUR field data collector with new data collection capabilities, new business logics, or support for new logging formats.

Input libraries:

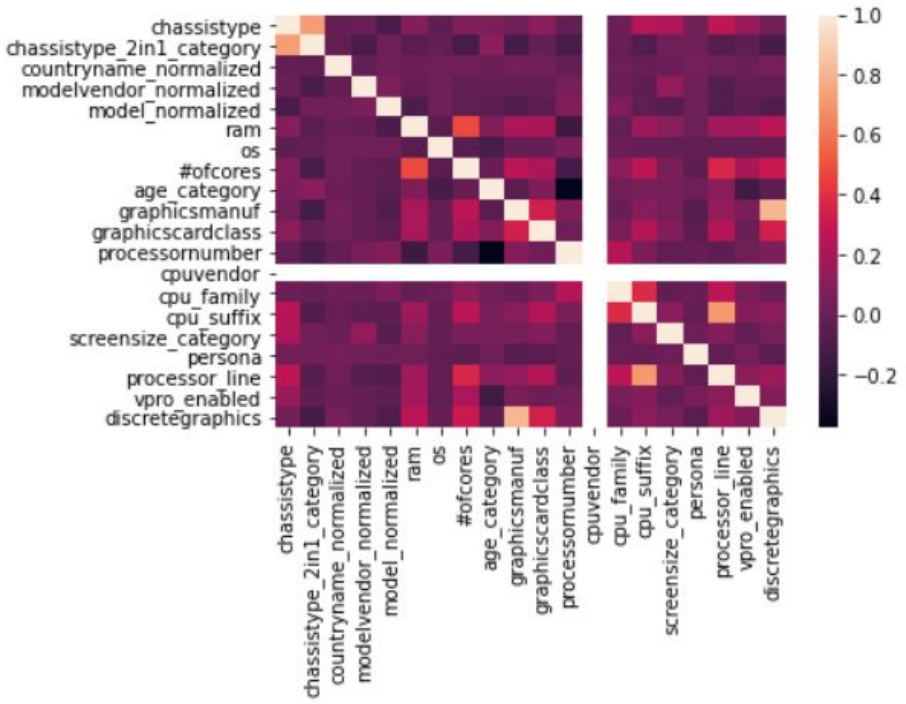
Input Libraries – or ILs – are the data collection units of the Intel SUR collector. The nature of an IL is to allow for strong code reuse, thus reducing TTM. Use of existing ILs allows customers to leverage the expertise of other teams. Over time, customers can gradually build up a custom collection of ILs.

ILs expose at least one input (or metric). However, ILs can reconfigure themselves at any time to change the number and the nature of their inputs without requiring a stop and restart of the collector. If needed, this allows for the development of complex modules with the ability to adapt to various platform changes.

2.Data Cleaning and EDA:

After we got the data, there were some nan and weird values. We cleaned the data and did some data transformation, like one-hot encodings on categorical features, so that we can do further tests and analysis.

When cleaning is done, we explore some important features we will use and see their distributions. We investigate some of the main features of the user's. See in appendix *2.1* to *2.5*.



3.Paired-T Tests On Numerical Features (RAM):

The goal of our hypothesis testing here is to check if there is a significant difference between the RAM of different types of users. The reason we use paired t-test is that two sample t-test is designed for testing differences between independent groups. We consider each persona as an independent group.

> Null hypothesis: There is no difference in RAM between the two types of personas (users).
>
> Alternative hypothesis: There is a difference in RAM between the two types of personas(users).
>
> Results: See all results in the appendix *3.1*. Because most of the p-values are significant, we extract the insignificant ones here.

	persona_pair	pvalue
0	(Office/Productivity, Win Store App User)	0.366362
1	(Casual Gamer, Gamer)	0.062092
2	(Entertainment, Casual User)	0.051284
3	(Entertainment, File & Network Sharer)	0.232316
4	(Communication, Casual User)	0.364350
5	(Communication, File & Network Sharer)	0.423373
6	(Casual User, Entertainment)	0.051284
7	(Casual User, Communication)	0.364350
8	(Casual User, File & Network Sharer)	0.839934
9	(Gamer, Casual Gamer)	0.062092
10	(Win Store App User, Office/Productivity)	0.366362
11	(File & Network Sharer, Entertainment)	0.232316
12	(File & Network Sharer, Communication)	0.423373
13	(File & Network Sharer, Casual User)	0.839934

From the p-values we get from all pairs of personas, we can conclude that most pairs of personas have a significant difference in RAM, which means we reject the null hypothesis and move to the alternative hypothesis. We also did the test for other numerical features and the results further proved our thoughts.

4.Chi-Square Tests for Independence on Categorical Features:

Our goal here is to find if the categorical features are related or independent from the personas (user types). A chi-square test for independence compares two variables in a contingency table to see if they are related. In a more general sense, it tests to see whether distributions of categorical variables differ from each other.

> Null hypothesis: The two categorical variables are independent (no association between the two variables)
>
> Alternative hypothesis: The two categorical variables are dependent (there is an association between the two variables)

The first step is to transform the features in our dataframe into crosstables. Use the *chassistype* as an example here.

chassistype	**2 in 1**	**Desktop**	**Intel NUC/STK**	**Notebook**	**Tablet**
persona					
Casual Gamer	365	3514	54	4939	0

Casual User	434	4414	402	5859	0
Communication	443	1720	241	2499	1
Content Creator/IT	289	1789	120	2820	0
Entertainment	144	1146	85	1978	0
File & Network Sharer	93	714	125	963	0
Gamer	320	3460	82	4770	3
Office/Productivity	562	1863	197	4059	0
Web User	2002	10109	662	18602	6
Win Store App User	365	1396	177	2486	1

After the transformation, we can apply the contingency table, and get the p-values between the persona. Using a threshold of 0.05, we can determine whether the p -value we get is significant or not. See all results in appendix *3.2*.

5.Machine learning models

There are three main points in our experiment setting:

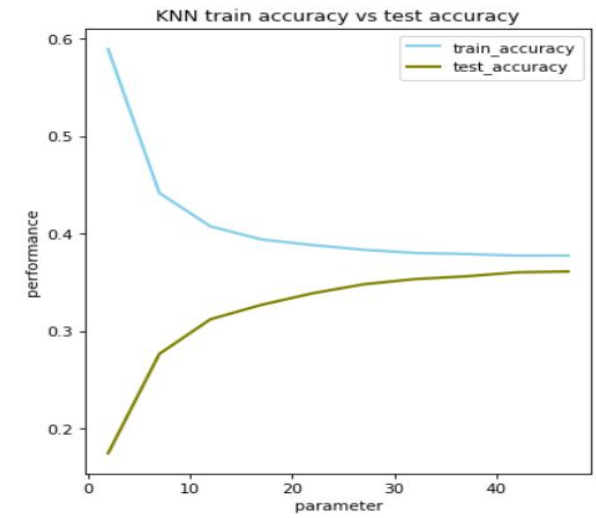
- Start 5 trials for the whole experiment. In each trail, generate random train and test data.
- For each classifier, try different parameters.
- Use both accuracy and F1 score as metric

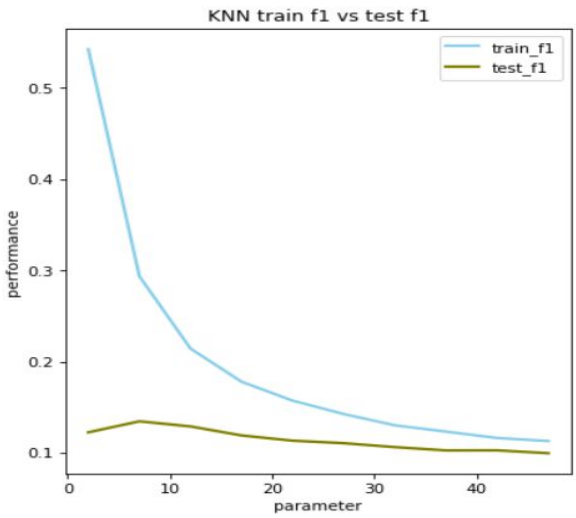
We mainly use 6 different machine learning models to predict the persona: K Nearest Neighbours, Decision Tree, Random Forest, Neural Network, Stochastic Gradient Descent and Logistic Regression. For each of the models, the first split the data into training and testing sets, and then fit the training data into our model. Also, optimizing parameters and trails We compare the models' performance using accuracy score and F1 score ($F1 = 2 \times \frac{precision*recall}{precision+recall}$).

Analysis

1.K Nearest Neighbours

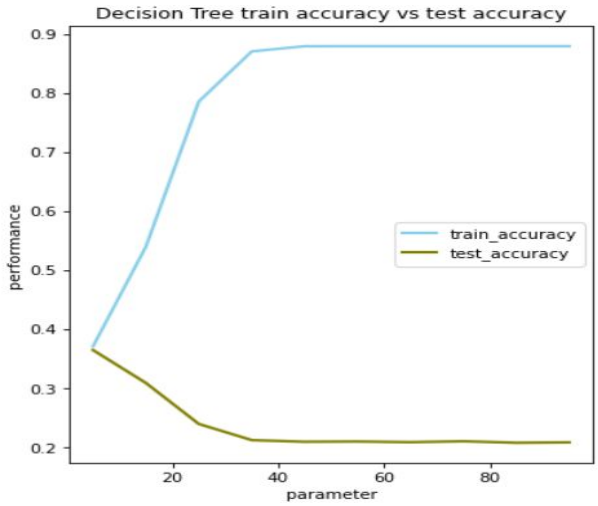
KNN is a model that classifies data points based on the points that are most similar to it. It uses test data to make an "educated guess" on what an unclassified point should be classified as. In our case, we are trying to predict a user's type by identifying which cluster the user is in. The following is the graph of performance. The parameter here is: N neighbours.

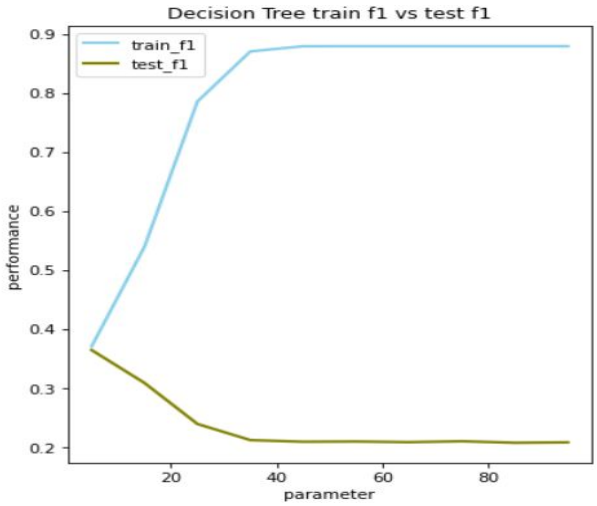




2.Decision Tree

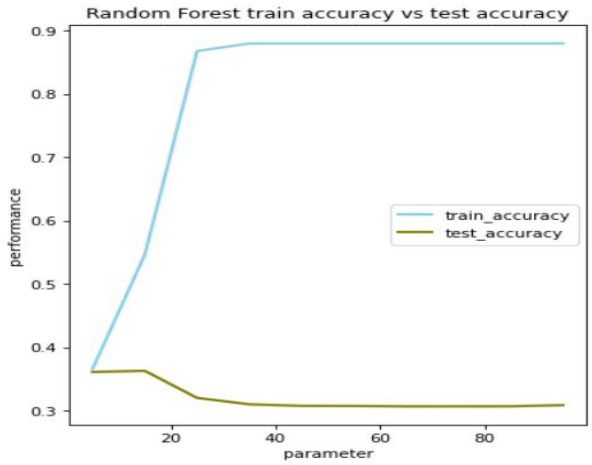
Decision trees are constructed via an algorithmic approach that identifies ways to split a data set based on different conditions. It is one of the most widely used and practical methods for supervised learning. The following is the graph of performance. The parameter here is: max_depth.

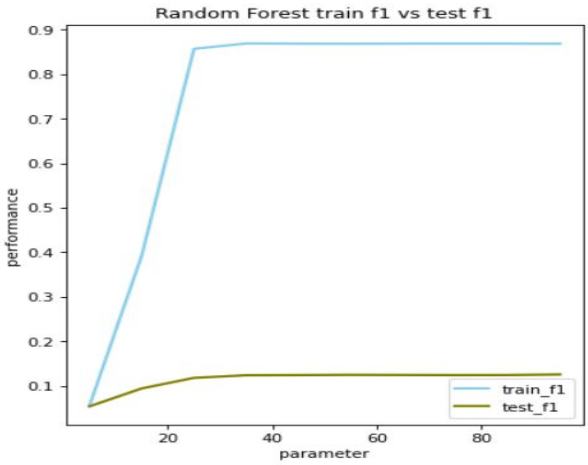




3.Random Forest

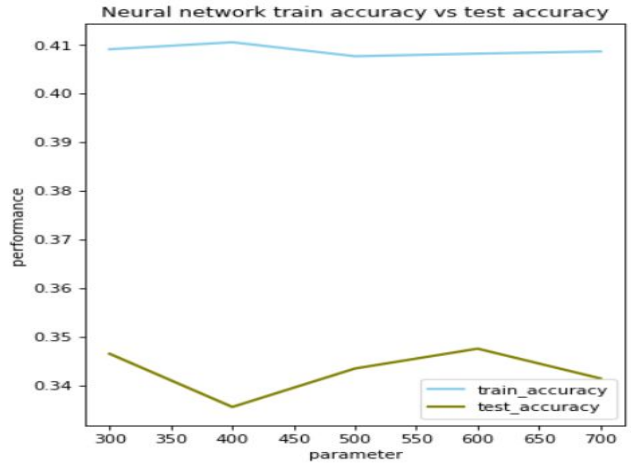
Based on decision trees, random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction. The following is the graph of performance. The parameter here is: max_depth.

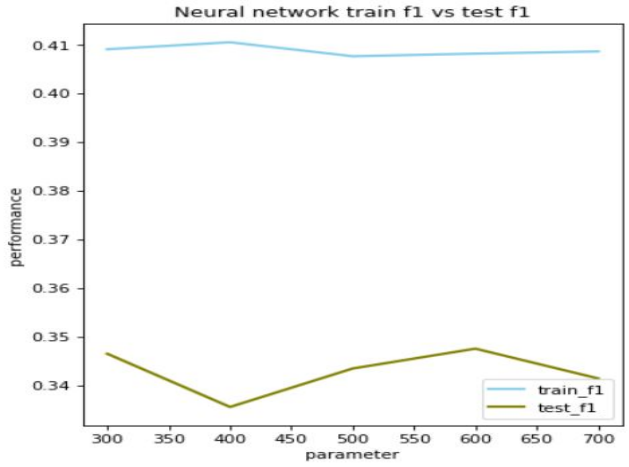




4.Neural Network --Multilayer Perceptron

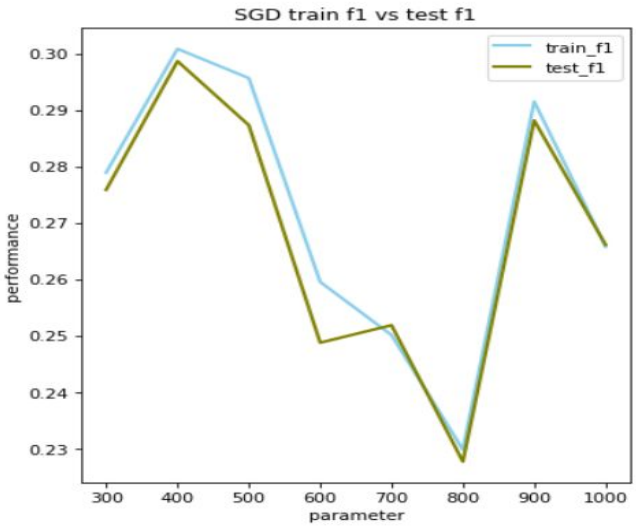
A perceptron is a simple binary classification algorithm. It helps to divide a set of input signals into two parts—“yes” and “no”. A multilayer perceptron (MLP) is a perceptron that teams up with additional perceptrons, stacked in several layers, to solve complex problems. The parameter here is: max_iteration.





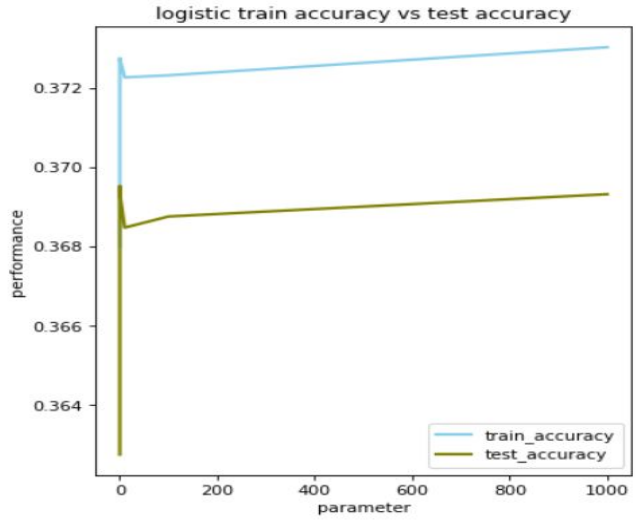
5.Stochastic Gradient Descent

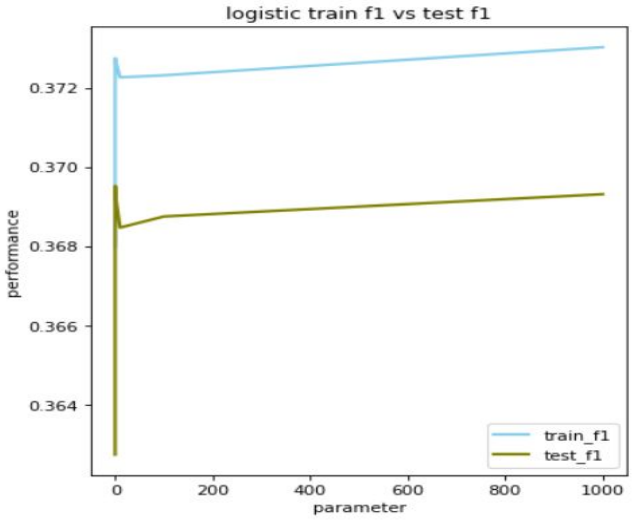
Gradient, in plain terms, means slope or slant of a surface. So gradient descent literally means descending a slope to reach the lowest point on that surface.The main goal of SGD is to reach the minimum of loss function. The parameter here is: max_iteration.



6.Logistic Regression

Logistic Regression is a Machine Learning algorithm which is used for classification problems, it is a predictive analysis algorithm and based on the concept of probability.





Conclusion

Based on our feature analysis and hypothesis testing, we conclude that most of our selected features are correlated with persona, except the CPU vendor. This may because the vendors are not a key factor of the user system setting, a vendor can have various kinds of product. In this case, it makes sense that CPU vendor is not a related feature with persona.

From the performance analysis of all six models, we conclude that the Neural Network -- Multilayer Perceptron is the best fit for our data. The first reason we consider is the flexibility. They are very flexible and can be used generally to learn a mapping from inputs to outputs. This flexibility allows them to be applied to other types of data.

Appendix:

1.1 (feature description)

features	Data Type	Description
load_ts	timestamp without time zone	Time the data collected
guid	character varying	Unique ID of the user
chassistype	character varying	Type of user's PC
countryname	character varying	User's country
modelvendor	character varying	Brand of the PC
modelvendor_normalized	character varying	Specific model of the PC
ram	double precision	Memory of the PC
os	character varying	Operating system of the PC
#ofcores	character varying	Number of cores in the CPU
age_category	character varying	The age range user in
graphicsmanuf	character varying	Brand of the graphic
gfxcard	character varying	Type of gfx card
graphicscardclass	character varying	Class of the graphic card
processornumber	character varying	The number of processors
cpuvendor	character varying	CPU manufacturer
cpuname	character varying	Name of the CPU
cpucode	character varying	Specific model of the CPU

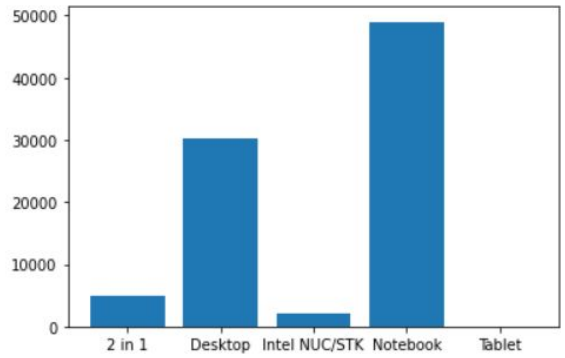
cpu_family	character varying	The category of the CPU
cpu_suffix	character varying	Type of the CPU
screensize_category	character varying	Screen size
persona	character varying	User type
firstreportdate	character varying	First report date
lastreportdate	character varying	Last report date
discretegraphics	character varying	If the PC has a discrete graphic card
cpu_stepping	character varying	Stepping of the CPU

1.2 (outputs of data collection)

	MEASUREMENT_TIME	ID_INPUT	VALUE	PRIVATE_DATA
	Filter	Filter	Filter	Filter
1	2020-11-21 07:03:28.346	0	1	0
2	2020-11-21 07:03:28.346	1	4294967295	0
3	2020-11-21 07:03:28.346	2	4294967295	0
4	2020-11-21 07:03:28.346	3	1	0
5	2020-11-21 07:03:28.346	4	0	0
6	2020-11-21 07:03:28.346	5	100	0
7	2020-11-21 07:03:29.346	0	1	0
8	2020-11-21 07:03:29.346	1	4294967295	0
9	2020-11-21 07:03:29.346	2	4294967295	0
10	2020-11-21 07:03:29.346	3	1	0
11	2020-11-21 07:03:29.346	4	0	0
12	2020-11-21 07:03:29.346	5	100	0
13	2020-11-21 07:03:30.345	0	1	0
14	2020-11-21 07:03:30.345	1	4294967295	0
15	2020-11-21 07:03:30.345	2	4294967295	0
16	2020-11-21 07:03:30.345	3	1	0
17	2020-11-21 07:03:30.345	4	0	0
18	2020-11-21 07:03:30.345	5	100	0
19	2020-11-21 07:03:31.344	0	1	0
20	2020-11-21 07:03:31.344	1	4294967295	0
21	2020-11-21 07:03:31.344	2	4294967295	0
22	2020-11-21 07:03:31.344	3	1	0
23	2020-11-21 07:03:31.344	4	0	0
24	2020-11-21 07:03:31.344	5	100	0

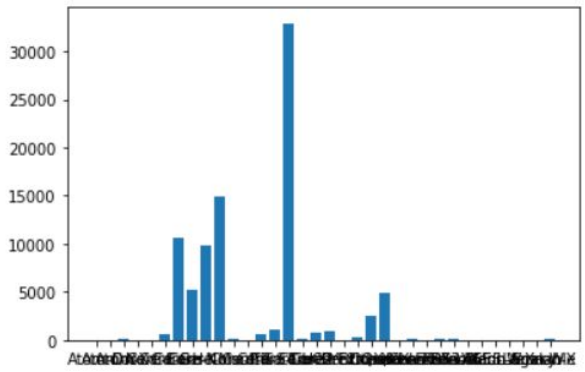
INPUT_NAME	INPUT_DESCRIPTION	INPUT_TYPE
Filter	Filter	Filter
BATTERY(0)	Battery: AC/DC	1
BATTERY(1)	Battery: BatteryLifeTime	1
BATTERY(2)	Battery: BatteryFullLifeTime	1
BATTERY(3)	Battery: BatteryFlag	1
BATTERY(4)	Battery: Status	1
BATTERY(5)	Battery: LifePercent	1

2.1 (data of chassistype)



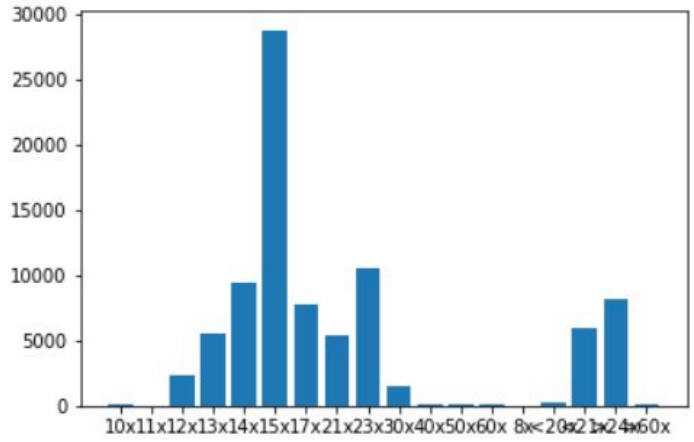
chassistype	
2 in 1	5017
Desktop	30125
Intel NUC/STK	2145
Notebook	48975
Tablet	11

2.2 (data of cpu suffix)



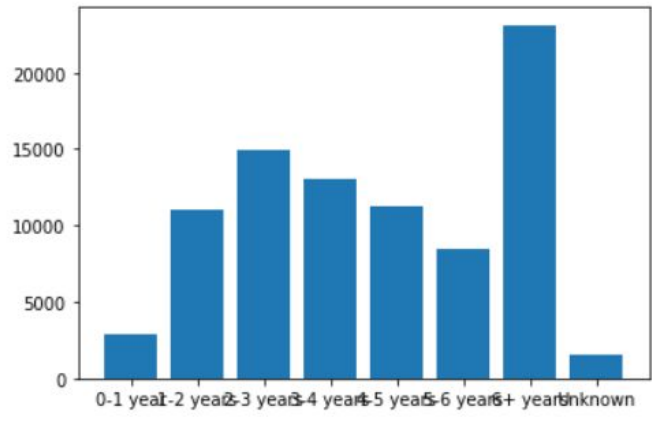
cpu_suffix	
Atom-D	6
Atom-N	6
Atom-X	131
Core-C	19
Core-E	5
Core-G	527
Core-H	10570
Core-K	5261
Core-M	9870
Core-No suffix	14927
Core-P	66
Core-R	46
Core-S	650
Core-T	1140
Core-U	32938
Core-X	159
Core-Y	729
Core2 Duo	866
Core2 Extreme	7
Core2 Quad	307
Other	2508
Pentium/Celeron	4952
Xeon-E	28
Xeon-E3	205
Xeon-E3-L	3
Xeon-E3-M	86
Xeon-E5	168
Xeon-E5-L	1
Xeon-E5-W	1
Xeon-L	1
Xeon-Legacy	25
Xeon-Scalable	5
Xeon-W	10
Xeon-X	50

2.3 (data of screen size)



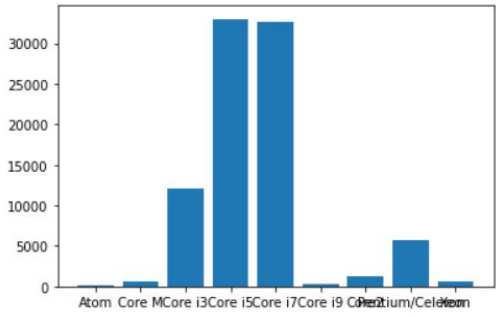
screensize_category	
10x	148
11x	9
12x	2331
13x	5588
14x	9401
15x	28732
17x	7753
21x	5403
23x	10555
30x	1472
40x	173
50x	86
60x	78
8x	41
<20x	217
<21x	6000
>24x	8167
>60x	119

2.4 (data of age)



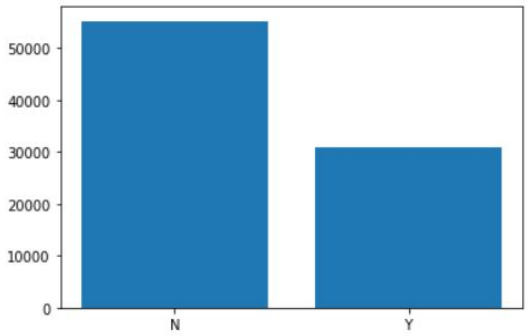
age_category	
0-1 year	2901
1-2 years	11066
2-3 years	14939
3-4 years	13077
4-5 years	11225
5-6 years	8431
6+ years	23122
Unknown	1512

2.5 (data of screen size)



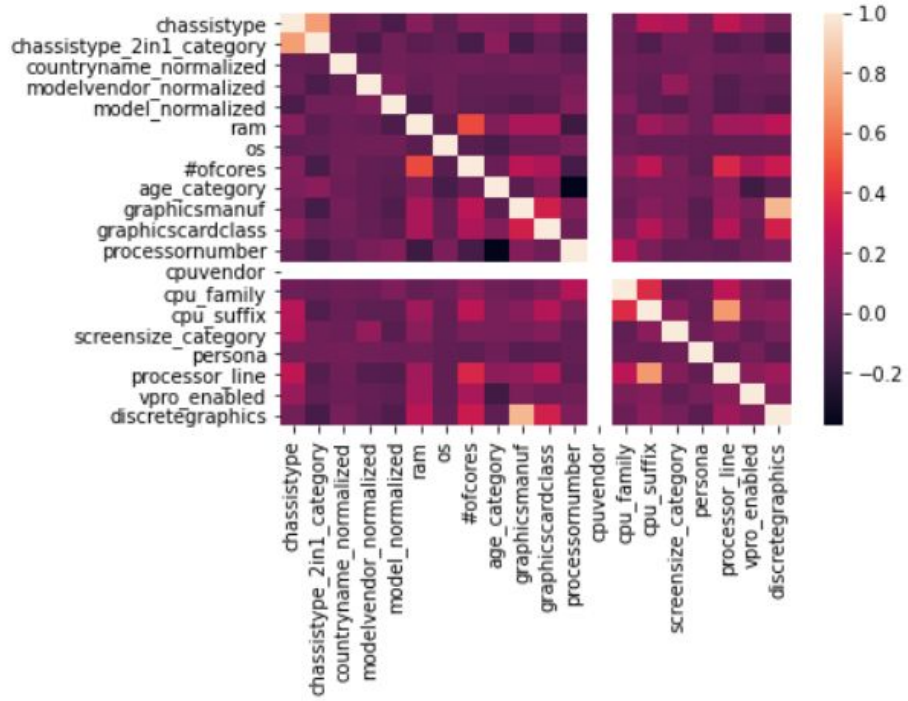
cpu_family	
Atom	143
Core M	598
Core i3	12055
Core i5	33020
Core i7	32619
Core i9	280
Core2	1232
Pentium/Celeron	5739
Xeon	587

2.4 (data of age Discrete Graphic)



discretegraphics	
N	55263
Y	31010

2.5 (Correlation Matrix And HeatMap)



3.1(results of paired-t testing)

	persona_pair	pvalue
0	(Web User, Office/Productivity)	3.766131e-06
1	(Web User, Casual Gamer)	5.583513e-63
2	(Web User, Entertainment)	4.226522e-02
3	(Web User, Communication)	2.253443e-11
4	(Web User, Casual User)	2.661729e-13
5	(Web User, Gamer)	2.102470e-48
6	(Web User, Content Creator/IT)	1.159044e-113
7	(Web User, Win Store App User)	8.051888e-07
8	(Web User, File & Network Sharer)	7.158699e-04
9	(Office/Productivity, Web User)	3.766131e-06
10	(Office/Productivity, Casual Gamer)	1.640917e-58
11	(Office/Productivity, Entertainment)	1.310209e-05
12	(Office/Productivity, Communication)	2.814926e-17
13	(Office/Productivity, Casual User)	2.031728e-17
14	(Office/Productivity, Gamer)	2.066233e-49
15	(Office/Productivity, Content Creator/IT)	5.970581e-89
16	(Office/Productivity, Win Store App User)	3.663619e-01
17	(Office/Productivity, File & Network Sharer)	3.672629e-08
18	(Casual Gamer, Web User)	5.583513e-63
19	(Casual Gamer, Office/Productivity)	1.640917e-58
20	(Casual Gamer, Entertainment)	3.397121e-13
21	(Casual Gamer, Communication)	1.708208e-07
22	(Casual Gamer, Casual User)	2.634763e-13
23	(Casual Gamer, Gamer)	6.209231e-02
24	(Casual Gamer, Content Creator/IT)	1.356447e-15
25	(Casual Gamer, Win Store App User)	3.063130e-50
26	(Casual Gamer, File & Network Sharer)	2.619122e-06
27	(Entertainment, Web User)	4.226522e-02
28	(Entertainment, Office/Productivity)	1.310209e-05
29	(Entertainment, Casual Gamer)	3.397121e-13

	persona_pair	pvalue
0	(Entertainment, Communication)	1.490570e-02
1	(Entertainment, Casual User)	5.128439e-02
2	(Entertainment, Gamer)	6.199759e-10
3	(Entertainment, Content Creator/IT)	6.322245e-29
4	(Entertainment, Win Store App User)	5.385314e-06
5	(Entertainment, File & Network Sharer)	2.323156e-01
6	(Communication, Web User)	2.253443e-11
7	(Communication, Office/Productivity)	2.814926e-17
8	(Communication, Casual Gamer)	1.708208e-07
9	(Communication, Entertainment)	1.490570e-02
10	(Communication, Casual User)	3.643502e-01
11	(Communication, Gamer)	1.286297e-04
12	(Communication, Content Creator/IT)	2.923503e-26
13	(Communication, Win Store App User)	1.798222e-16
14	(Communication, File & Network Sharer)	4.233734e-01
15	(Casual User, Web User)	2.661729e-13
16	(Casual User, Office/Productivity)	2.031728e-17
17	(Casual User, Casual Gamer)	2.634763e-13
18	(Casual User, Entertainment)	5.128439e-02
19	(Casual User, Communication)	3.643502e-01
20	(Casual User, Gamer)	1.606087e-08
21	(Casual User, Content Creator/IT)	8.274969e-40
22	(Casual User, Win Store App User)	6.812619e-16
23	(Casual User, File & Network Sharer)	8.399339e-01
24	(Gamer, Web User)	2.102470e-48
25	(Gamer, Office/Productivity)	2.066233e-49
26	(Gamer, Casual Gamer)	6.209231e-02
27	(Gamer, Entertainment)	6.199759e-10
28	(Gamer, Communication)	1.286297e-04
29	(Gamer, Casual User)	1.606087e-08

	persona_pair	pvalue
0	(Gamer, Content Creator/IT)	7.949237e-22
1	(Gamer, Win Store App User)	2.902713e-43
2	(Gamer, File & Network Sharer)	1.884330e-04
3	(Content Creator/IT, Web User)	1.159044e-113
4	(Content Creator/IT, Office/Productivity)	5.970581e-89
5	(Content Creator/IT, Casual Gamer)	1.356447e-15
6	(Content Creator/IT, Entertainment)	6.322245e-29
7	(Content Creator/IT, Communication)	2.923503e-26
8	(Content Creator/IT, Casual User)	8.274969e-40
9	(Content Creator/IT, Gamer)	7.949237e-22
10	(Content Creator/IT, Win Store App User)	1.126696e-73
11	(Content Creator/IT, File & Network Sharer)	1.804335e-17
12	(Win Store App User, Web User)	8.051888e-07
13	(Win Store App User, Office/Productivity)	3.663619e-01
14	(Win Store App User, Casual Gamer)	3.063130e-50
15	(Win Store App User, Entertainment)	5.385314e-06
16	(Win Store App User, Communication)	1.798222e-16
17	(Win Store App User, Casual User)	6.812619e-16
18	(Win Store App User, Gamer)	2.902713e-43
19	(Win Store App User, Content Creator/IT)	1.126696e-73
20	(Win Store App User, File & Network Sharer)	1.109398e-08
21	(File & Network Sharer, Web User)	7.158699e-04
22	(File & Network Sharer, Office/Productivity)	3.672629e-08
23	(File & Network Sharer, Casual Gamer)	2.619122e-06
24	(File & Network Sharer, Entertainment)	2.323156e-01
25	(File & Network Sharer, Communication)	4.233734e-01
26	(File & Network Sharer, Casual User)	8.399339e-01
27	(File & Network Sharer, Gamer)	1.884330e-04
28	(File & Network Sharer, Content Creator/IT)	1.804335e-17
29	(File & Network Sharer, Win Store App User)	1.109398e-08

3.2(results of chi-square testing)

Feature pairs	***p-values***
(persona, chassistype)	*6.570130e-296*
(persona, chassistype_2in1_category)	*3.452896e-106*
(persona, countryname_normalized)	*0.000000e+00*
(persona, modelvendor_normalized)	*0.000000e+00*
(persona, model_normalized)	*0.000000e+00*

(persona, os)	*1.287536e-26*
(persona, age_category)	*0.000000e+00*
(persona, graphicsmanuf)	*0.000000e+00*
(persona, graphicscardclass)	*0.000000e+00*
(persona, cpuvendor)	*1.000000e+00*
(persona, cpu_family)	*5.001971e-190*
(persona, cpu_suffix)	*0.000000e+00*
(persona, screensize_category)	*0.000000e+00*
(persona, processor_line)	*0.000000e+00*
(persona, vpro_enabled)	*1.054635e-226*
(persona, discretegraphics)	*0.000000e+00*