**노트북의 가성비 평가를 위한 데이터 분석**

홍슬기(19971027), 장현수(1997????)

[aimedaca47@gmail.com](mailto:aimedaca47@gmail.com)

**서론**

대학생 3학년. 슬슬 노트북 수명이 다해가는 시기이다. 노트북을 구매하려고 하는데, 카페에서 흔하게 보이는 대기업 노트북밖에 모른다. CPU? GPU? 노트북 크기? 무게? 브랜드? 누군가는 게임을, 누구는 서류작업을, 그래픽작업을, 코딩을, 소비자마다 원하는 것이 다르기 때문에, 개개인이 원하는 사양에 맞춰서 합당한 가격의 노트북을 고르기란 무척 어려운 일이다. 따라서 각 요인들이 노트북 가격에 미치는 영향에 대해 분석해보고, 구매하려는 특정 노트북 제품 사양을 보고 가격을 추측(prediction)해보기 위해 분석을 시작했다.

노트북 제품에 관한 여러가지 정보를 보고, 데이터 분석를 수행하여 통계적으로 가격을 예측해보자. 게다가 이 관계를 잘 학습할 수 있다면 주어진 노트북 정보에 대해 가격 예측을 수행하여, 실제로 책정된 가격이 비싼 편인지도 판단할 수 있을 것이다.

**분석에 사용할 Dataset** : Laptop Prices <https://www.kaggle.com/ionaskel/laptop-prices>

View가 1k 이상이며, 꽤 많은 사람들이 download했고 이미 해결법이 나와있지는 않은 dataset을 선정하였다. Discussion 정보는 없어서 온전히 자력으로 분석을 수행하였다.

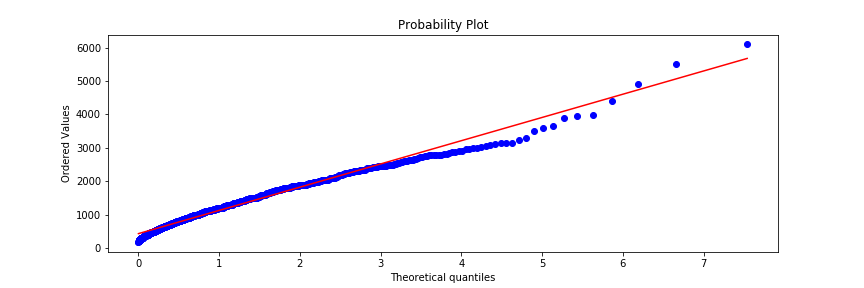
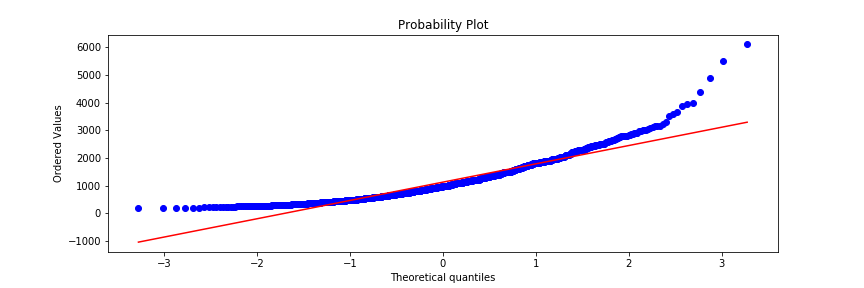
Dataset의 크기는 1303\*13이다. Column 하나는 index정보라서, 실질적으로 이용할 수 있는 variable은 12개라고 봐야한다.

**Column 정보**

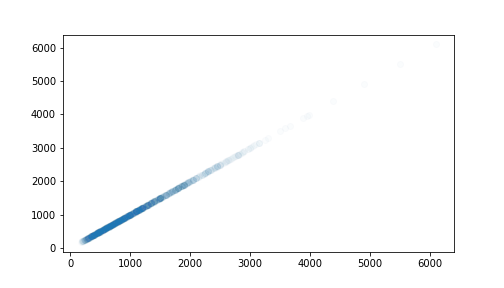
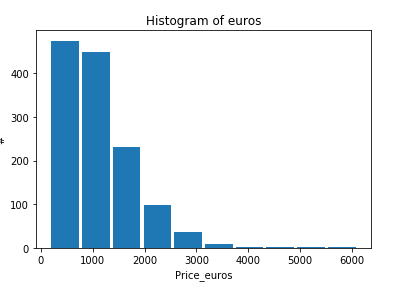
1. Company : 회사명. 총 19가지이다.
2. Product : 618가지 제품명이다.
3. TypeName : Laptop Type이다. 6가지이다.
4. Inches : 스크린 인치 수
5. ScreenResolution : 해상도. 40가지.
6. CPU : 118가지. String.
7. RAM : 9가지이다. XXGB 형태로 적혀있다.
8. Memory : 39가지. SSD와 HDD의 조합으로 이루어져 있다.
9. GPU : 110가지. String.
10. OpSys : OS 관련 정보이다. 운영체제 미포함의 경우도 있다. 9가지.
11. Weight : 노트북 무게이다. 뒤에 kg단위가 붙어있는 string 형태.
12. Price\_euros : 노트북의 가격이다. 이를 잘 예측하는 것이 데이터 분석의 최종 목적이다.

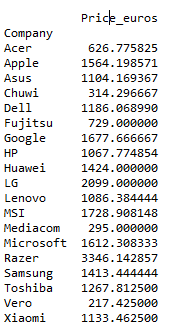
**가격(Price\_euros) 분포 확인하기**  
: Column 분석 이전에, 가격의 분포를 먼저 확인해보았다.

* Q-Q graph를 통한 확인 : 좌측은 norm, 우측은 exponential이다.



* 산포도 확인을 위한 Scatter graph와 histogram : alpha를 0.02로 두었다. 그래프의 아랫부분에 몰려있다.



* 따라서, 노트북의 가격은 고르게 분포하는 것이 아니라 주로 2000 euro 이하에 분포하며, 그 이상의 가격은 드문드문 커져서 지수형 분포와 비슷하게 나타나는 것을 확인할 수 있었다.
* 또한, Company별 가격도 상당한 차이가 난다. 

**각 Column의 분석 방향 및 분포 확인**  
: 단순한 숫자형 데이터가 아닌 column이 다수이므로, 각 column data에 대한 preprocessing이 필요하다. 그러기 위해서는 데이터의 어떤 부분을 중점적으로 볼 것인지에 대한 계획이 필요하다.

그리고 조건부확률처럼 변인통제를 해야하므로, column별로 관측데이터가 가장 많은 값을 미리 알아두기로 했다.

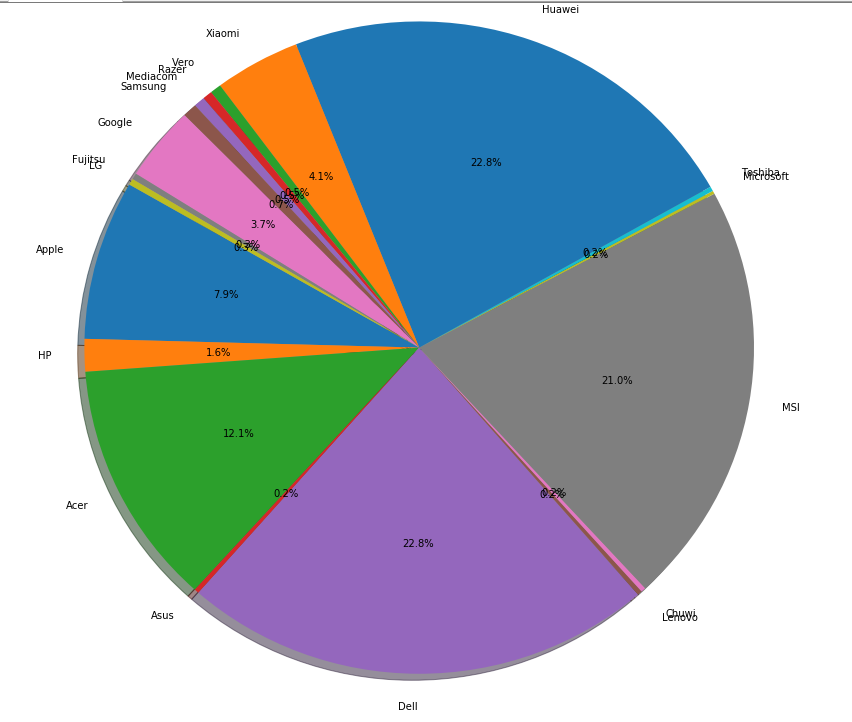
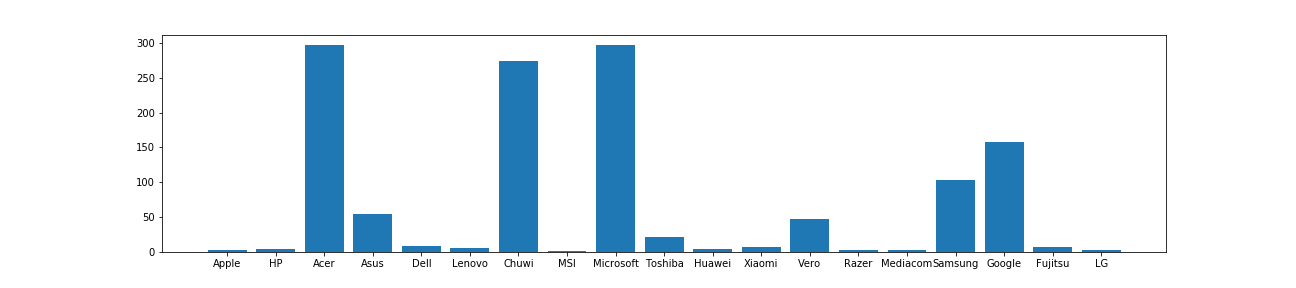
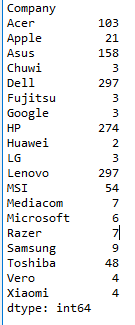
Column의 값끼리 우선순위 관계가 있는 경우는 파란색으로 나타냈고, 단순히 카테고리 역할만 하는 것은 주황색으로 표시했다.

* **Company** : 앞으로의 모든 분석을 전체 dataset에 대해서만 할 게 아니라, 19가지 각 company별로 따로 확인할 수 있을 것이다. 이것은 기존 dataset을 분류하는 category로서의 의미가 있을 것이다.

1. **종류**

'Apple' 'HP' 'Acer' 'Asus' 'Dell' 'Lenovo' 'Chuwi' 'MSI' 'Microsoft' 'Toshiba' 'Huawei' 'Xiaomi' 'Vero' 'Razer' 'Mediacom' 'Samsung' 'Google' 'Fujitsu' 'LG'

1. **종류 별 개수**



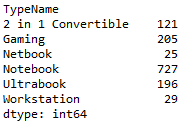
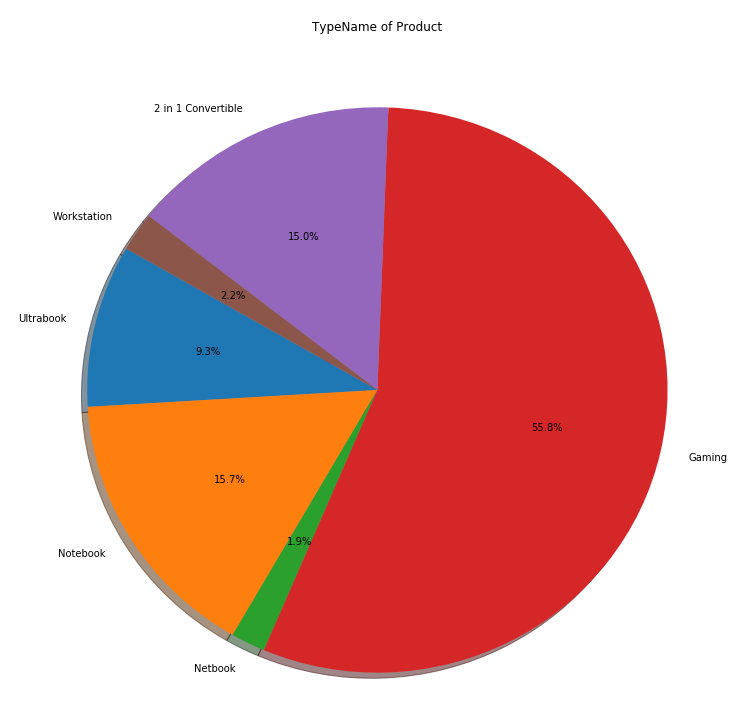
**[그림] Company당 dataset내 제품 개수 통계**

* Product : 회사가 이미 정해져있기 때문에, 모델마다 세분화해서 분석하는 경우라면 사용가치가 있을 것이다. 같은 Apple 제품이어도 MacBook Pro, MacBook Air등으로 나뉘기 때문이다. 하지만 618가지나 존재하므로, 분석에서 우선순위를 두지는 않을 것이다.
* **TypeName** : Notebook(56%), Gaming(16%), 그 외 4가지가 29%로, 총 6가지로 나누어져 있으므로 company와 비슷하게 category로 사용할 것이다. 사용자마다 노트북 사용의 목적이 다르므로 중요한 정보이다. 예를 들면, 게이밍 노트북의 특징으로 gpu나 해상도가 좋은 경우가 많다는 식의 통계를 내거나, 게이밍 노트북의 전반적인 가격은 어떤 요소로 인해 나눠지며 가격대는 어느정도로 예상해야할지 등의 통계를 낼 것이다.

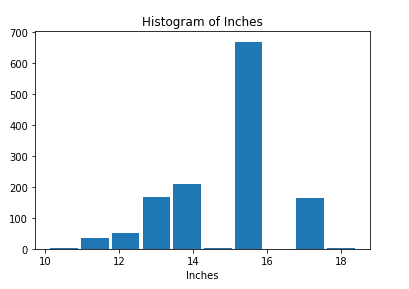
1. **종류**

'Ultrabook' 'Notebook' 'Netbook' 'Gaming' '2 in 1 Convertible' 'Workstation'

1. **종류 별 개수**

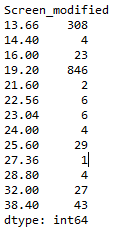
 

* **Inches** : 숫자형 데이터이다. 다만 inches와 가격을 그냥 비교하면 다른 요인의 영향을 받게되기 때문에, 변인 통제를 잘 해야한다. 실제로 모든 inches와 price 사이의 correlation을 계산하면 이라서 큰 차이가 보이지 않는다. 다른 요인들을 고정시켜서 특정 row만 선택하여 분석할 것이다.

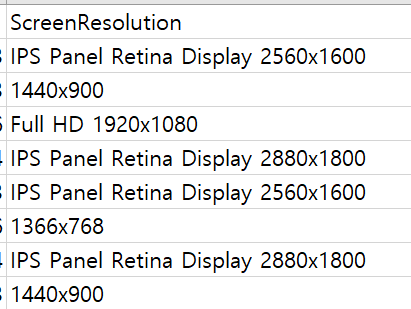
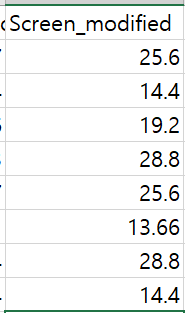


**[그림] inches의 노트북 수 분포 히스토그램**

* **ScreenResolution** : width\*height 외에도 다른 디스플레이 정보가 붙어있는 경우도 존재한다. 이 부분은 누락시키고, 해상도에서 픽셀수를 수치화하여 고해상도와 저해상도를 구분하도록 할 것이다. Width\*height를 그냥 사용하면 값의 차이가 아주 커서 원하는 분석에 있어 오차가 생길 수 있으므로, **양수값으로 rank**를 매길 것이다. 이렇게 preprocessing이 끝나면, inches형과 같은 방법으로 분석할 것이다.

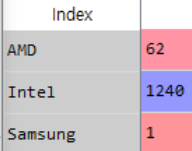
****

**[그림] 해상도 정보 분포 확인 (수치화)**

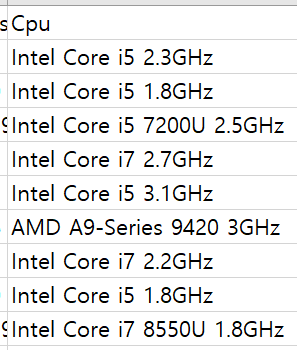
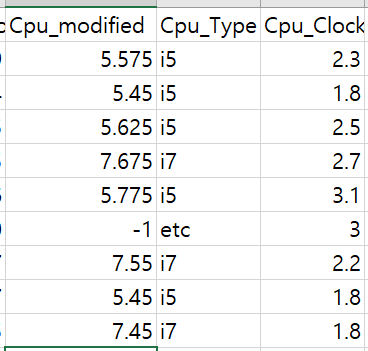
 🡪 

**[그림] 적절한 공식을 통한 해상도 데이터의 수치화**

* **CPU** : CPU 성능 또한 raw data는 string형이지만, 등급이 존재한다. 따라서 rank를 매기는 preprocessing이 필요하다. Intel이 절대다수이므로 i3, i5, i7 등으로 크게 등급을 매긴 후, 뒤에 따라오는 첫 숫자로 세대를 계산하여 우선순위를 추가계산 할 것이다. CPU는 inches와 해상도와 달리 가격과 상관관계가 클 것으로 추측되지만, 마찬가지로 변인 통제를 해서 분석하는 것 또한 시도할 것이다.

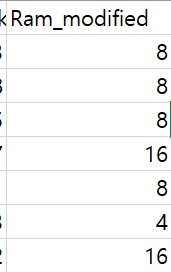


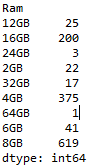
[그림] intel이 대부분이므로, 나머지는 etc로 분류. i3, i5, i7 등 등급과 클럭 정보로 적절히 수치화하였다.

 **🡪**  

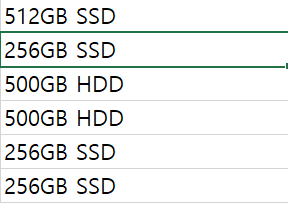
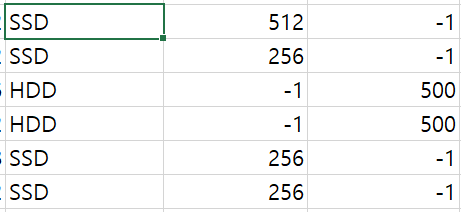
**[그림] CPU data 수치화하기**

* **RAM** : 9가지이고 4, 8GB가 제일 많다. 우선 string형에서 뒤에 ‘GB’를 뺀다. 따로 처리는 하지 않고 숫자만 남겨도 가격과 램의 상관관계가 로 계산되었으므로 그대로 사용할 것이다.

 **[그림] String에서 숫자형으로 바꾼 데이터**

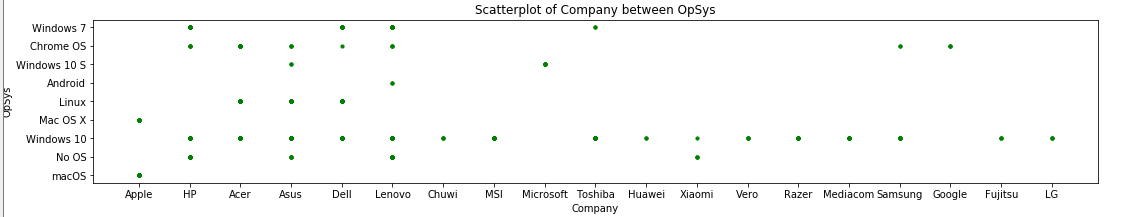
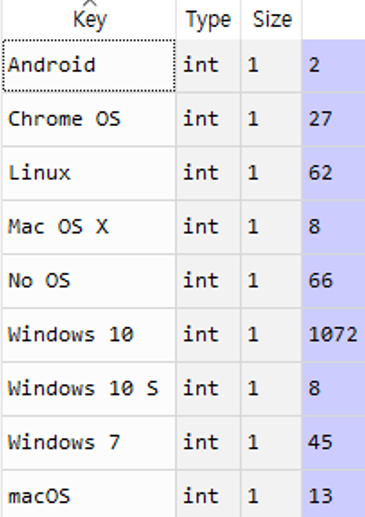
 **[그림] RAM 분포**

* **Memory** : HDD와 SSD의 조합으로 인해 생각보다 많은 분류가 있는데, SSD의 여부와 용량 등을 고려하는 식의 분석이 필요하다고 판단했다. 특히 하드디스크는 크게 비싸지 않기 때문에, **SSD의 포함여부와 용량만을 고려하여 숫자형태로** preprocessing 할 것이다.

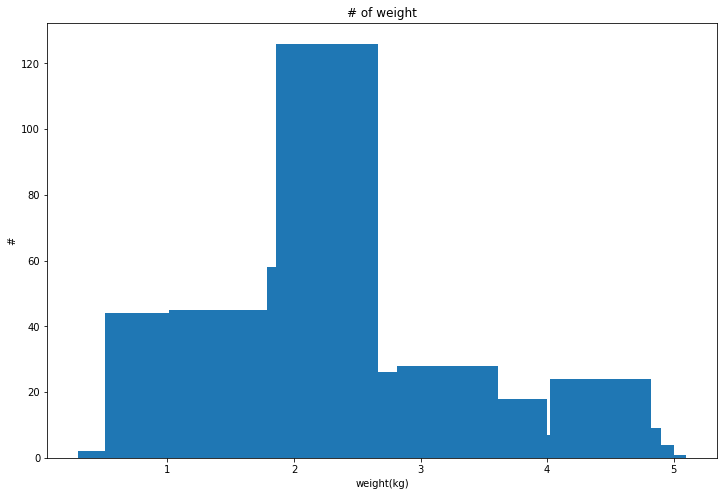
 🡪 

**[그림] Memory Data를 수치화. SSD를 중점적으로 고려하였다.**

* **GPU** : CPU와 마찬가지로 rank를 계산하여 수치형 데이터로 바꿀 것이다. 문제는, 종류가 너무 많아서 CPU처럼 간단하게 처리할 수 없다. 그래서 우선적으로 종류를 크게 분류해보았다.
* OpSys : window가 극단적으로 많기 때문에, 단순히 운영체제 포함과 미포함으로만 나눠서 볼 것이다. 포함여부가 가격에 얼마나 영향을 미치는지에 대해 조사할 것이다.



* **Weight** : RAM과 비슷하게, kg 단위만 버리고 숫자값을 그대로 사용할 것이다.



**[그림] 무게 정보 분포**

* **Price\_euros : 이 값이 예측하고자 하는 값이다.** 다른 요소들이 가격과 얼마나 관계가 있을지를 분석할 것이다. 직접적인 상관관계를 구할 수도 있고, 다른 조건을 고정시킨 다음에 비교할 수도 있다.

이렇게 dataset 정보에 대한 기본적인 통계와 전처리가 끝났다.

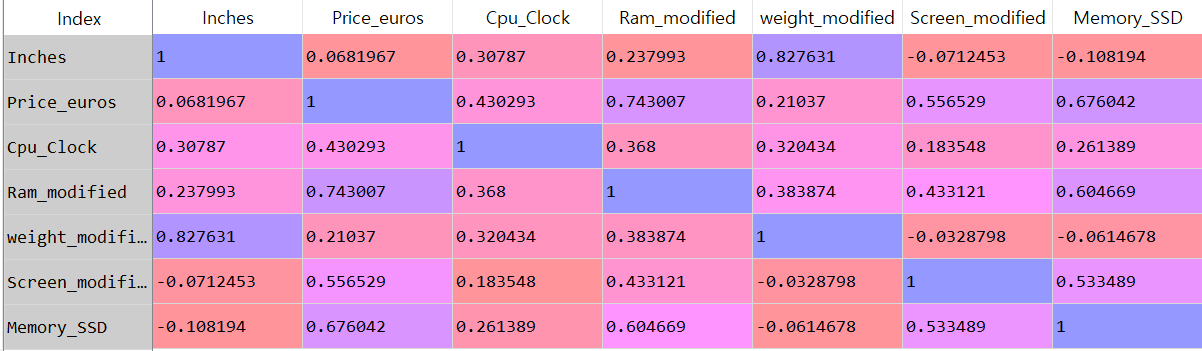
**가장 Common한 Data의 종류  
:** 앞으로의 분석에 이용하기 위하여 정리한 정보이다. 괄호 안의 정보는 도수이다.

|  |  |
| --- | --- |
| Company | Dell(297) , HP(274) , Lenovo(297) |
| TypeName | Notebook(727) |
| Inches | 15.6(665) |
| ScreenResolution (수치화 된 것) | 19.20(846) |
| Cpu | i5(423) |
| Ram | 8GB |
| OS | Windows 10 |

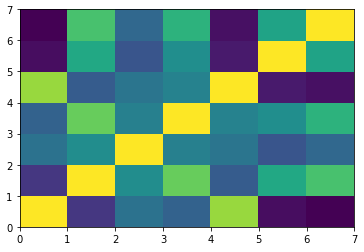
위 정보는 나중에 다른 column을 고정해놓고 특정 column끼리의 변화관계만 관측할 경우에 필요하다. 왜냐하면,이 데이터셋은 한 column내에서 특정 값이 압도적으로 많은 경우가 대부분이므로, 역으로 변수의 특정한 값에 해당하는 데이터가 너무 적은 경우는 올바른 관측이 어렵기 때문이다.

**Column간의 관계 분석  
:** column간의 관계에 대한 분석 결과이다. 이 결과는 전체 dataset에 대하여 수행했다. 특정 부분에 주목하기 전, variable간의 관계에 있어서 전체적인 흐름을 파악하기 위함이다. (GPU 고려시 다시 구해서 적용할 것.)

* Correlation : 인덱스는 두 이미지 모두 같다.



**[그림] Correlation Matrix**



**[그림] HeatMap을 이용한 Visualization**

**(아래는 수정한거 없음! 걍 남겨둔 것 뿐)**

**우리가 할 것 생각한 것.**

일단 파란색으로(숫자형으로 나타낼 수 있는 것) 표시한, 가격에 영향을 줄 수 있는 요인이 7가지야.

그리고 가격이랑의 상관관계도 봐야하니까, 7+1하면 8개의 column으로 상관관계 행렬 구해보자.

이건 굉장히 단순한 관계이지. 요인 고정한것도 없고, 그냥 전체 dataset에서 구하는거니까, 다른 요인에 의한 노이즈 또한 있겠지. (그래도 1차적으로 이런 분석은 필요하다고 생각해)

이 다음부터는 그냥 내 생각인데, 만약 각 요인의 변화에 의해 가격을 **선형적**으로 추측이 가능하면, w1x1 + w2x2 + w3x3+ …. = 가격 이런 느낌이겠지? 그렇다면 요인 1개의 변화를 제외하고 나머지는 모두 고정한 상태에서 ㅂ교해야할거야. 무슨 말이냐면, 예시를 들어보자.

Ex) 램 4기가, cpu 등급 5.0, gpu등급 4.5, 무게 1.0~1.5kg(무게는 너무 세세하게 차이나니까) 인치(이건 굳이 고정 안해도 될것같긴 한데) 등등… 7가지 숫자형 데이터들 중에서 1가지(예를들면, 메모리)만 바꿀 수 있게 하고, 나머지를 고정하자. 그러면 다른 요인에 전혀 방해받지 않는 상태에서 메모리와 가격사이의 관계만 추측 가능하겠지?

그게 w1인거지. (통계학이랑 실험에서 변인통제를 생각해보자.) 다른 속성들도 마찬가지로, **하나빼고 다 고정하는식으로 가격 변화를 1대1로 관측**한다면 가격을 얼마만큼이나 바꿀 수 있는지에 대한 값이 숫자(weight)로 나오겠지?  
(물론 선형적이지 않을 수 있어. 그래서 아까 rank의 규모를 고민한거였어. Cpu 한단계 차이나는걸 딱 1만 차이나게 해도 될지…)

이런 숫자값들(weight)을 전부 구한 다음, 각 7가지 column에 대한 weight를 비교하는 도표를 그려서 가격에 미치는 영향력의 크기를 비교해보자. 대략 이런 게 데이터 분석이지 않을까? 그리고 더 나아가서, xi값들에 실제 input 요인값 (아까 그 램 8기가, 무게 이런 것 7가지) 입력해서 실제 가격과의 오차도 구할 수 있는거야! 말하자면 가격 예측과 실제값의 오차를 계산하는거지.