

Keras with R

데스크톱 환경 구축 (우분투 가상 머신)

김연우

2019-01-03

차례

1. Virtual box 설치
2. 가상 머신에 Ubuntu 설치
3. Ubuntu 및 가상 머신 설정
4. Keras 실행을 위한 사전 설치
5. R 및 Keras package 설치
6. Keras 테스트
7. 문제해결

참고. Keras with GPU

1. Virtual box 설치

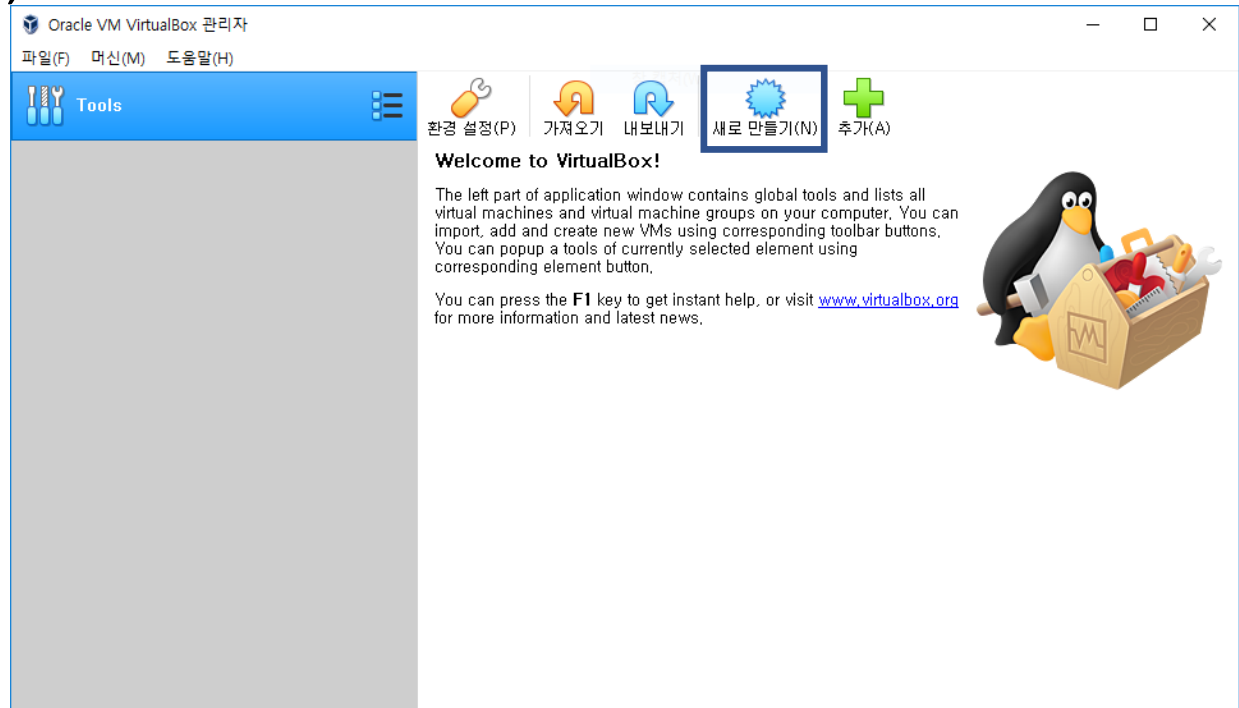
- VirtualBox 다운로드
 - <https://www.virtualbox.org/wiki/Downloads>
 - VirtualBox 6.0.0 platform packages 선택
 - Windows hosts 선택
- 설치
 - 설치 프로그램 실행 후 옵션 변경 없이 계속 Next 클릭

2. 가상 머신에 Ubuntu 설치

- Ubuntu Desktop 이미지 다운로드
 - <https://www.ubuntu.com/download/desktop>
 - Ubuntu 18.04.1 LTS Download 선택
 - 페이지 이동 후
ubuntu-18.04.1-desktop-amd64.iso 자동 다운로드

2. 가상 머신에 Ubuntu 설치

- VirtualBox에 가상머신 생성 (1/4)
 - ‘새로 만들기’
클릭

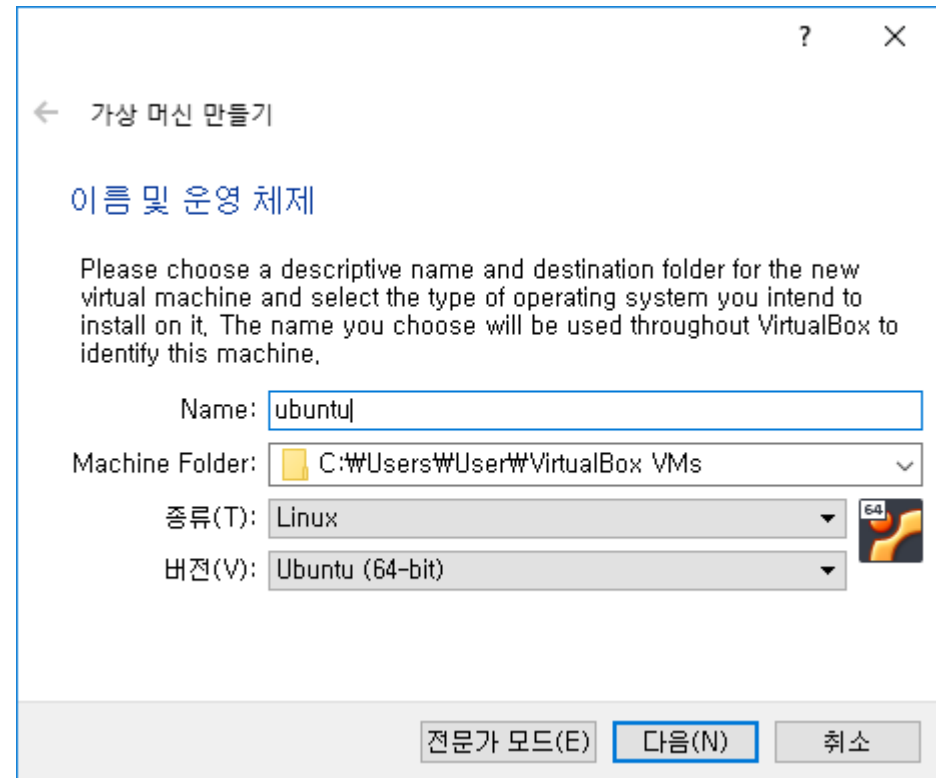


2. 가상 머신에 Ubuntu 설치

- VirtualBox에 가상머신 생성 (2/4)

- 이름 및 운영체제

- Name: 원하는 이름으로
 - Machine folder: 기본 값으로
 - 종류: Linux
 - 버전: Ubuntu (64-bit)

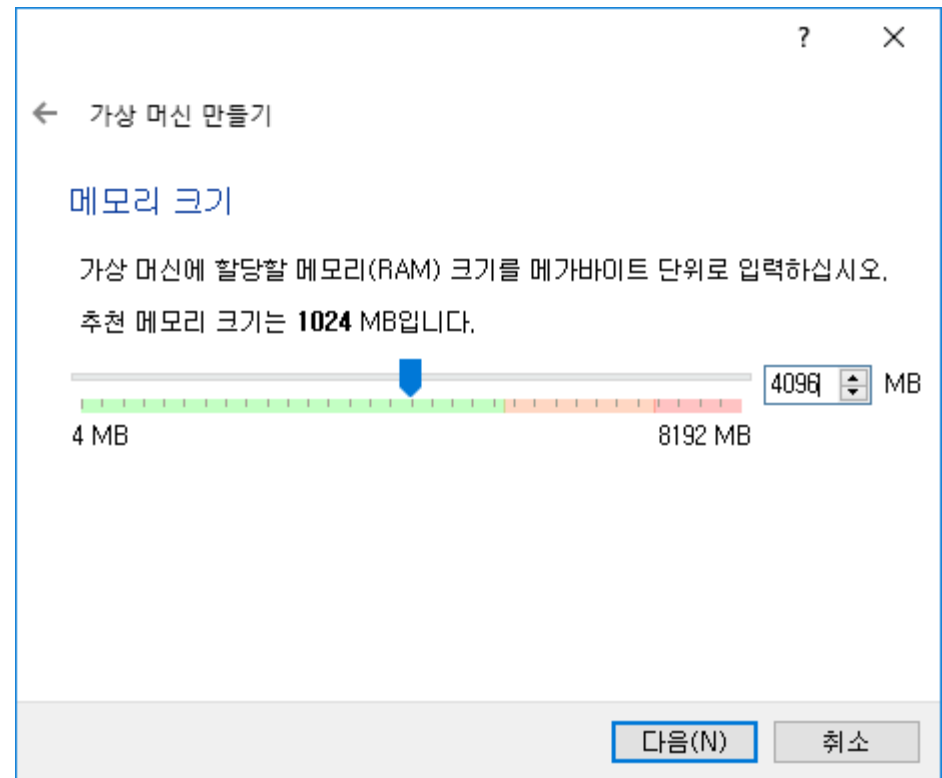


2. 가상 머신에 Ubuntu 설치

- VirtualBox에 가상머신 생성 (3/4)

- 메모리 크기

- 안전범위(연두색) 안에서 지정
 - 최대 RAM 사이즈의 절반 정도 추천



2. 가상 머신에 Ubuntu 설치

- VirtualBox에 가상머신 생성 (4/4)
 - 하드디스크
 - '지금 새 가상 하드 디스크 만들기' 선택
 - 하드디스크 파일 종류
 - 'VDI' 선택
 - 물리적 하드 드라이브에 저장
 - '동적 할당' 선택 ('고정 크기' 선택도 괜찮음)
 - 파일 위치 및 크기
 - 32GB 이상

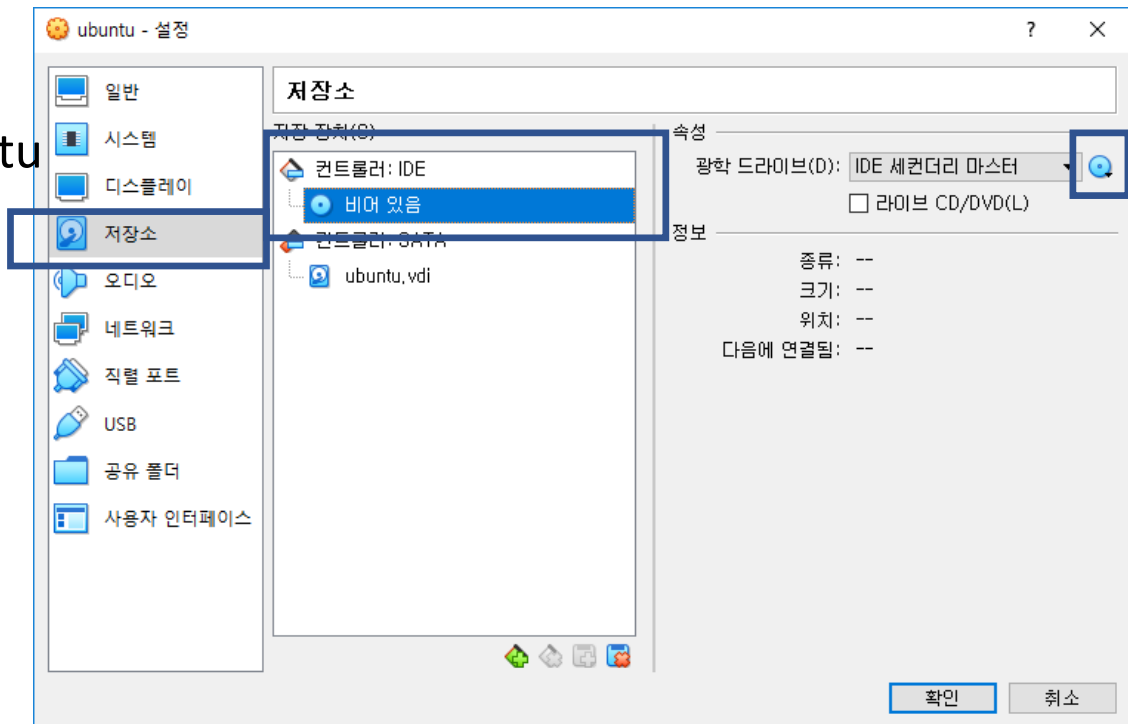
2. 가상 머신에 Ubuntu 설치

- 가상머신에 Ubuntu 설치 이미지 삽입 (1/2)
 - 생성한 가상 머신 클릭 > '설정' 클릭



2. 가상 머신에 Ubuntu 설치

- 가상머신에 Ubuntu 설치 이미지 삽입(2/2)
 - '저장소' 클릭 > 컨트롤러: IDE 아래 시디 모양 클릭
 - 오른쪽 IDE 세컨더리 마스터 우측의 시디 모양 클릭
 - '가상 광디스크 파일 선택'
 - 다운 받은 Ubuntu iso 파일 선택
 - '확인' 클릭



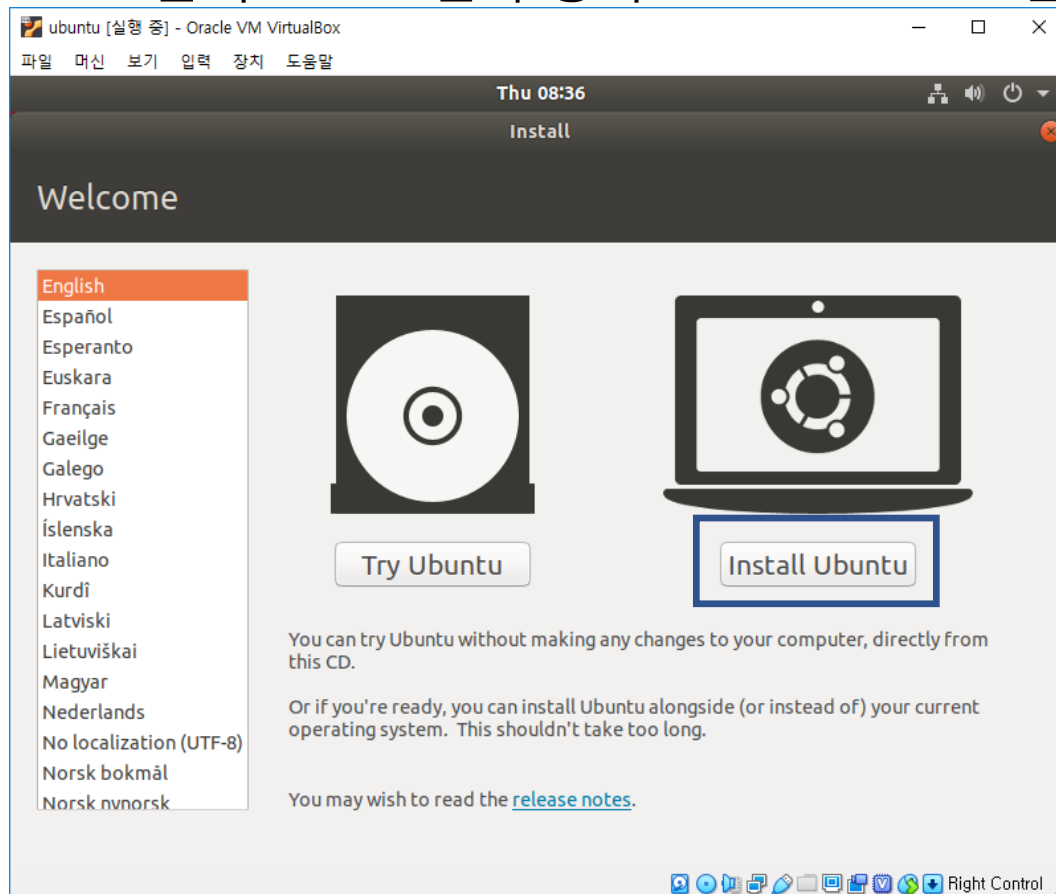
2. 가상 머신에 Ubuntu 설치

- 가상머신에 Ubuntu 설치 (1/4)
 - 생성한 가상 머신 클릭 > '시작' 클릭



2. 가상 머신에 Ubuntu 설치

- 가상머신에 Ubuntu 설치 (2/4)
 - Ubuntu 설치 프로그램 구동 후 'Install Ubuntu' 클릭



2. 가상 머신에 Ubuntu 설치

- 가상머신에 Ubuntu 설치 (3/4)
 - Keyboard layout
 - English (US) - English (US) 선택
(설치 후 한글 키보드로 재설정)
 - Update and other software
 - 'Normal installation' 선택
 - 'Download updates while installing ubuntu'
(빠른 설치를 위한다면 해제하고 설치 후 업데이트)
 - 'install third-party software'
(체크 하는 것 추천)

2. 가상 머신에 Ubuntu 설치


- 가상머신에 Ubuntu 설치 (4/4)
 - Install Type
 - 'Erase disk and install ubuntu' 선택
 - 'install now' > 'continue'
 - Where are you?
 - 'seoul'
 - Who are you?
 - user name 및 passwd 지정 > 'continue'
 - 재시작 및 남은 업데이트 진행

2. 가상 머신에 Ubuntu 설치

- 가상 머신에서 Ubuntu 구동이 느릴 경우
 - Linux mint (Xfce) 혹은 Xubuntu 배포판으로 대체 가능
 - 설치 과정은 동일
 - <https://linuxmint.com/download.php>
 - <https://xubuntu.org/download/>


3. Ubuntu 및 가상 머신 설정

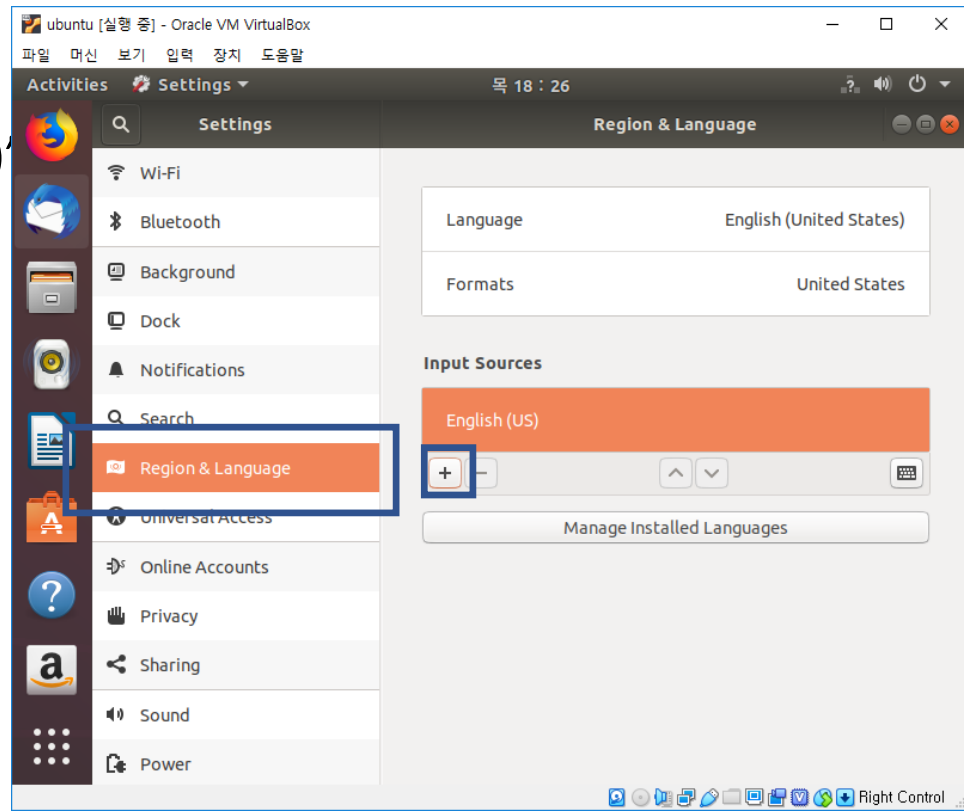
- 한국어 패키지 설치

- 윈도우키  클릭 후 검색 창에 'language support' 입력 하여 실행
- 자동으로 실행되는 업데이트 진행 후 'install / remove languages' 선택 > 'korean'에 체크
(설치과정에서 지역을 'seoul'로 설정했다면 이미 설치 되어 있음)
- 재부팅 (한글 입력 설정을 위해 반드시 해야함)

3. Ubuntu 및 가상 머신 설정

- 한글 입력기 설치 (1/2)

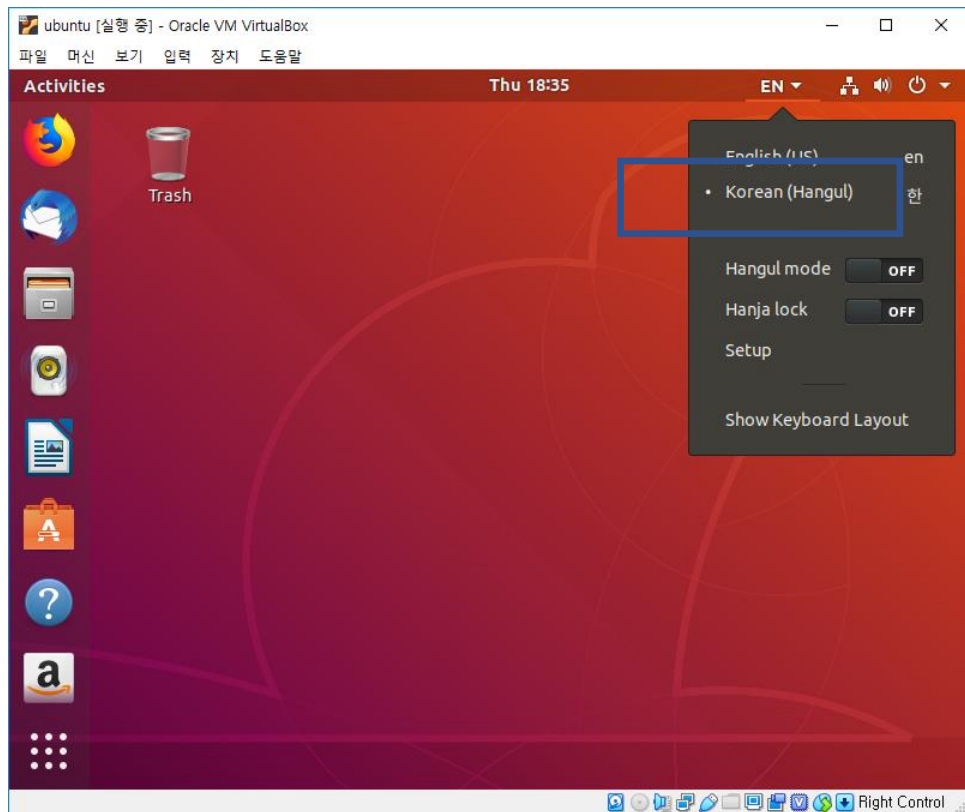
- 윈도우키  클릭 후 검색 창에 'setting' 입력 하여 실행
- 'Region and language' > Input Sources에서 '+' 클릭
- 'Korean' 선택
- 이후 선택지 'Korean(hangul)' 선택



3. Ubuntu 및 가상 머신 설정

- 한글 입력기 설치 (2/2)

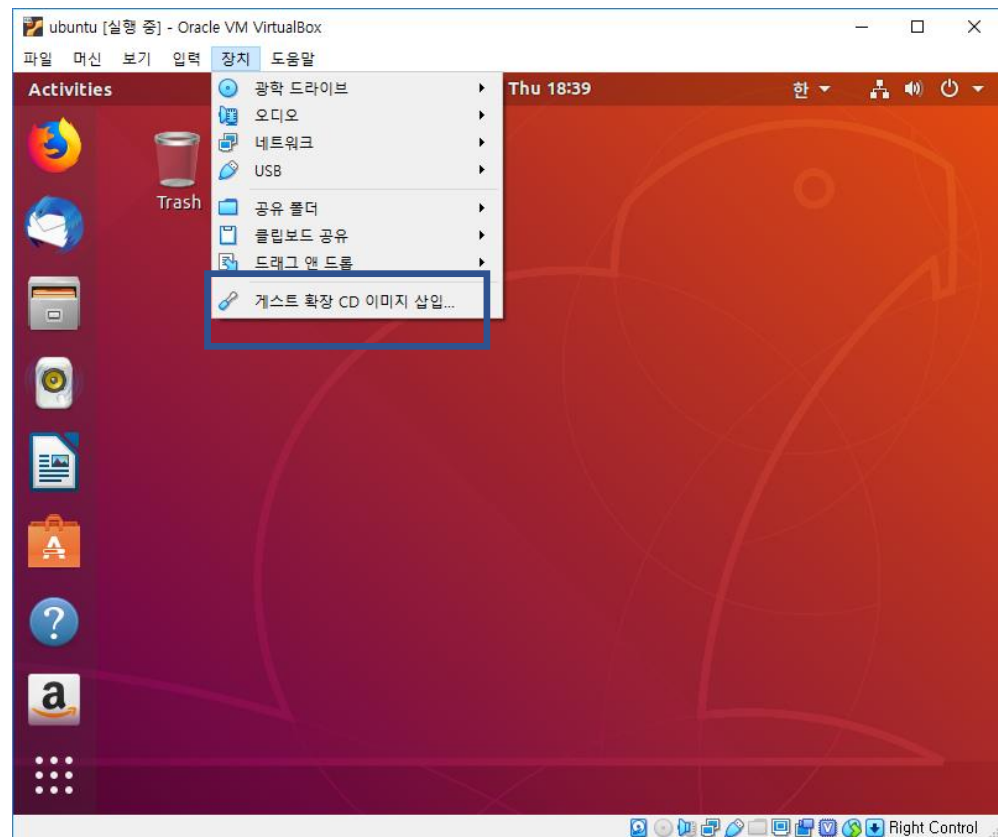
- 상단 상태표시줄에서 'EN' 클릭 하고 'Korean(Hangul)'로 변경
- '한/영' 키 이용하여 한글 모드 on/off가능



3. Ubuntu 및 가상 머신 설정

- Vbox 게스트 확장 설치

- 가상 머신 메뉴에서 '장치' > '게스트 확장 CD 이미지 삽입' 클릭
- 'Run' 클릭
- 이후 창 크기 변경 시 해상도가 자동으로 변경



3. Ubuntu 및 가상 머신 설정

- 가상 머신과 클립보드 공유

- 가상 머신 메뉴에서 '장치' > '클립보드 공유' > '양방향' 클릭

- 가상 머신과 폴더 공유

- 가상 머신 메뉴에서 '장치' > '공유 폴더' > '공유 폴더 설정' 클릭
- '머신 폴더' 우클릭 > '공유폴더추가'
- 폴더 경로 지정
- '자동마운트', '항상사용하기'에 체크
- virtual box 재실행

4. Keras 실행을 위한 사전 설치

- BLAS library (for modeling with CPU)
 - 터미널(Ctrl + Alt + T)에서

```
$ sudo apt-get update
$ sudo apt-get upgrade
$ sudo apt-get install python-pip python-dev
$ sudo apt-get install build-essential cmake git unzip
$ sudo apt-get install pkg-config libopenblas-dev liblapack-dev
```

- virtualenv (for TensorFlow on R)
 - 터미널에서

```
$ sudo apt-get install python-virtualenv
```

5. R 및 Keras package 설치

- R 설치
 - 터미널에서

```
$ sudo apt install r-base
```

- RStudio 설치
 - 직접 다운로드 후 설치
 - <https://www.rstudio.com/products/rstudio/download/#download>
 - 'RStudio 1.1.463 - Ubuntu 16.04+/Debian 9+ (64-bit)' 선택
 - 'rstudio-xenial-1.1.463-amd64.deb' 실행

5. R 및 Keras package 설치

- Keras package 설치
 - RStudio에서

```
> install.packages("keras")  
> library(keras)  
> install_keras()
```

6. Keras 테스트

- 테스트
 - RStudio에서
 - 다음 코드 실행

```
library(keras)
model <- keras_model_sequential() %>%
  layer_dense(units = 32, input_shape = c(784)) %>%
  layer_dense(units = 32)
```

- 아래와 같이 실행되면 성공

```
> model <- keras_model_sequential() %>%
+ layer_dense(units = 32, input_shape = c(784)) %>%
+ layer_dense(units = 32)
Using TensorFlow backend.
> |
```


7. 문제 해결

- 이슈: 'Old version of cryptography ([1, 2, 3]) may cause slowdown'
 - 터미널에서 다음 코드 실행

```
> sudo pip install --upgrade cryptography  
> sudo python -m easy_install --upgrade pyOpenSSL
```

- RStudio에서 Keras package 재설치

```
> install.packages("keras")  
> library(keras)  
> install_keras()
```

참고. Keras with GPU (주의! GPU 이용할 때만 실행)

- 주의! 가상 머신에는 설치 금지
- **cuda v9.0 and cudnn 7.2**
 - 아래 링크의 쉘스크립트 실행
 - R 이용 시 전체 실행
 - RStudio 이용 시 47~49행 삭제하고 실행
 - <https://gist.github.com/Mahedi-61/2a2f1579d4271717d421065168ce6a73>
 - 환경 변수 설정 (RStudio 이용 시)
 - '~/.profile' 파일에

```
export CUDA_HOME=/usr/local/cuda
export LD_LIBRARY_PATH=${LD_LIBRARY_PATH}:${CUDA_HOME}/lib64
PATH=${CUDA_HOME}/bin:${PATH}
export PATH
```

참고. Keras with GPU (주의! GPU 이용할 때만 실행)

- Keras GPU package
 - RStudio에서

```
> library(keras)  
> install_keras(tensorflow = "gpu")
```