Git & Github
EDA

2020.1.28.

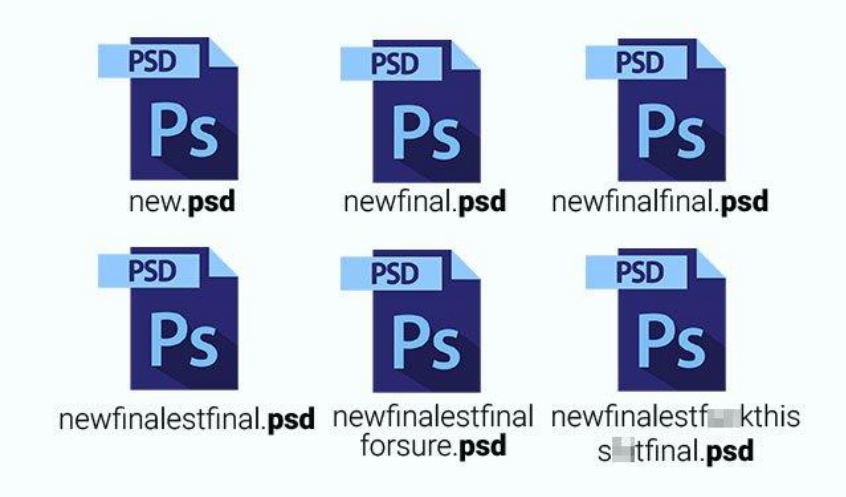
간정현, 이해환

목차

1. Git & GitHub 이해하기

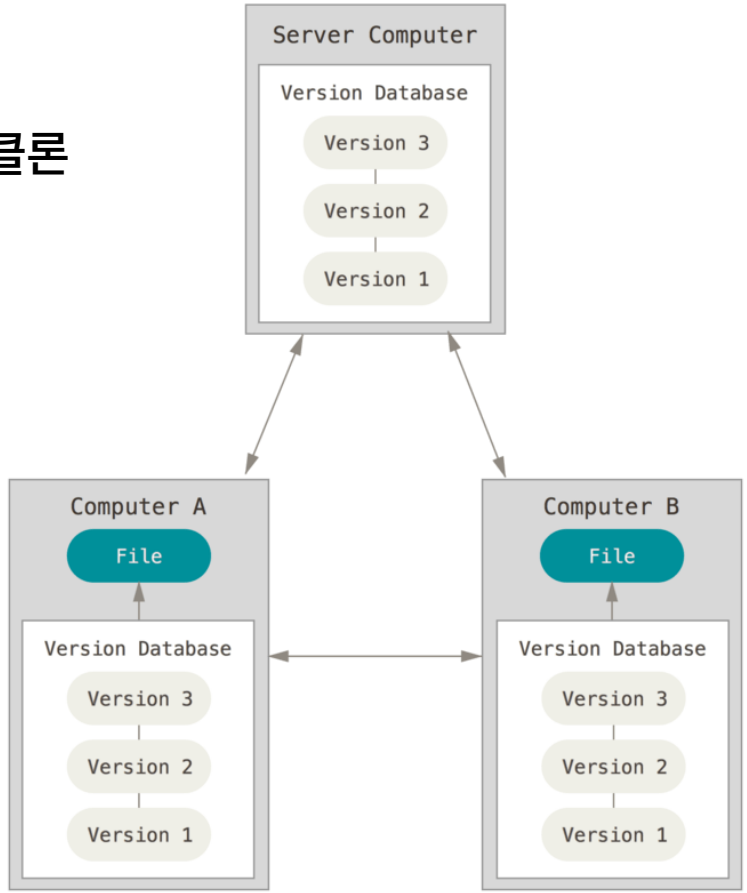
Git의 본질: 버전 관리 시스템

- 특정 디렉토리 안의 파일들을 감시
- 변경 내역을 기록 : 누가, 언제, 무엇을, 어떻게 수정했는가?



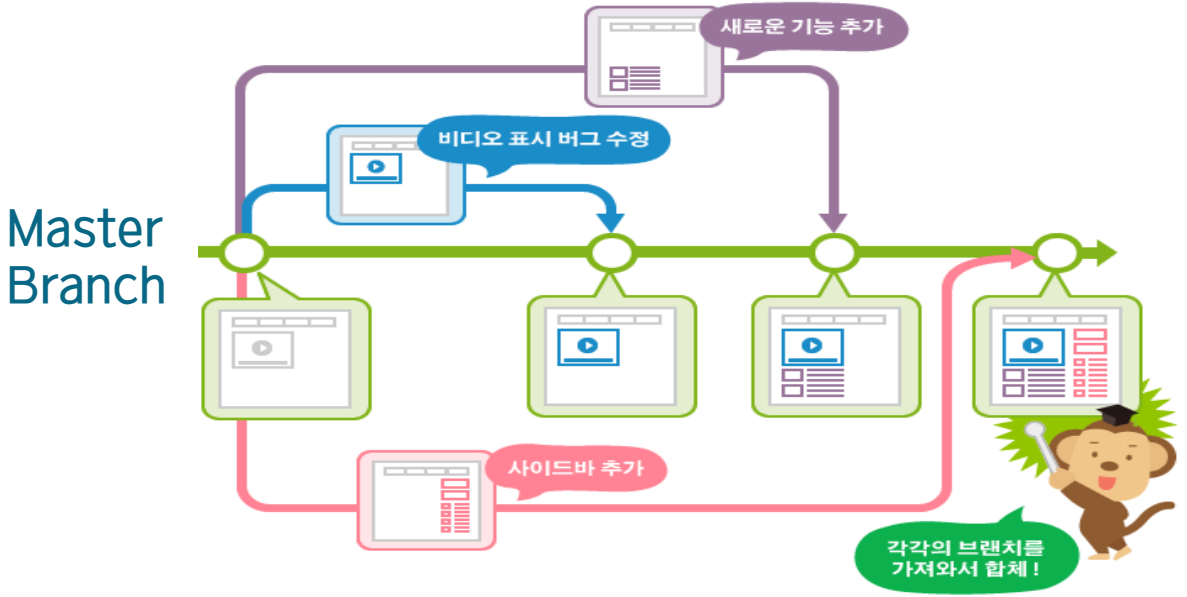
분산형 버전 관리 시스템

- 서버 저장소 & 히스토리를 모두 클론
- 서버와 로컬의 동기화!



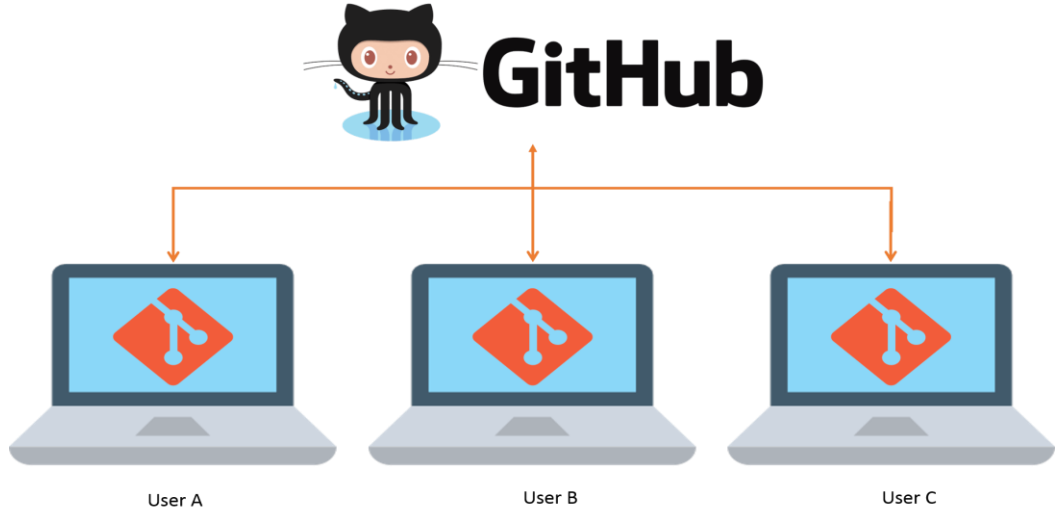
Branch

- 프로젝트에서 진행되는 여러 작업을 동시에 관리
- Master 브랜치를 중심으로 작업 가지치기 & 병합

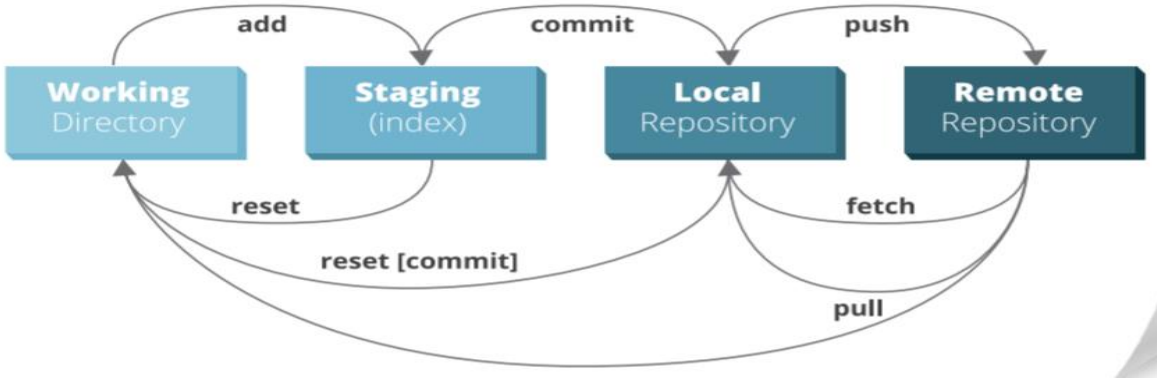


GitHub

- 원격 저장소 제공, 즉 협업의 중심 역할
- 수많은 오픈소스 프로젝트가 GitHub을 통해서 진행
- 누구나 공개 프로젝트에 기여 가능



Git & GitHub 작업 흐름



명령어	기능
git clone <repository>	GitHub 저장소를 로컬에 클론
git pull	GitHub 저장소의 업데이트를 로컬에 반영
git add <filename>	파일의 변경사항을 스테이지
git commit –m “<message>“	스테이지된 변경사항을 모두 업데이트
git push	로컬의 커밋을 GitHub에 반영

2. 실습: GitHub 으로 협업하기 (2~3인 1조)

	따라할 일!	뭘 하는건가요?
1	깃허브 저장소 생성 Settings > Collaborators 등록	조장님이 협업의 중심이 될 깃허브 저장소를 생성하고 협업할 조원분들에게 초대장을 보내주세요! 초대장은 이메일로 발송됩니다.
2	Git bash 실행	깃 명령어를 사용할 수 있는 명령창을 엽니다. 물론 CMD, Powershell 등에서도 깃 명령어를 사용할 수 있습니다.
3	$ git config --global user.name <yourName> $ git config --global user.email <yourName>	깃에게 사용자의 이름과 이메일 주소를 알려줍니다. 누가 언제 무엇을 어떻게 바꿨는지 추적하기 위해서 필수적인 과정입니다.
4	$ git clone <repository> $ cd <yourDirectory>	깃허브에 생성된 저장소를 내 컴퓨터에 복제합니다. Clone or Download 에서 저장소 주소를 복사해주세요. 복제한 후에는 해당 디렉토리로 이동해줍니다. 디렉토리 이름은 저장소 이름과 같습니다.
5	README.md 파일 편집, 저장	조장님이 README.md 파일을 수정해주세요! README.md 파일은 깃허브 저장소에서 가장 먼저 보여지는 파일입니다. 프로젝트에 대한 개요, 설명서 등을 적어주면 됩니다.
6	$ git add README.md	README.md 파일의 변경사항을 깃에 스테이징합니다. 모든 파일의 변경사항을 스테이징하려면 git add * 을 사용하면 됩니다.

	따라할 일!	뭘 하는건가요?
7	$ git commit –m “<message>”	스테이징된 변경사항을 모두 적용합니다. 로컬에서 실질적인 업데이트가 이루어지는 파트라고 생각하시면 됩니다.
8	$ git push origin master	로컬의 업데이트 사항을 깃허브에 반영합니다. 만약 깃허브에서 다른 사용자에 의한 업데이트가 먼저 있었다면 git pull 을 사용해서 깃허브의 업데이트를 로컬에 반영한 후 다시 푸시해줍니다.
9	$ git pull	조장님에 의해 깃허브의 최신 상태가 업데이트되었으므로, 이를 조원분들의 로컬 컴퓨터에 반영합니다.
10	$ notepad model.py	조원 한 분이 로컬에서 아무 코드나 생성 후 저장해주세요. 어떤 파일이든 상관 없습니다. 데이터 프로젝트라면 머신러닝 모델이 될 수도 있고, 분석 과정을 담은 코드일 수도 있을 것입니다.

11	$ git add model.py
	$ git commit –m “<message>”
	$ git push origin master
		아까와 같이 변경사항을 스테이징하고, 확정하고, 깃허브에 푸시합니다.

12	9 ~ 11의 반복	프로젝트가 끝날 때까지 pull > add > commit > push 반복!

3. 실습: GitHub 페이지 만들기

Jekyll

- 블로그 지향의 정적 사이트 생성기
- GitHub 에서 블로그 호스팅(마크다운 기반)

Jekyll

- Jekyll themes 에서 다양한 지킬 테마 확인 가능
- 깃허브에 업로드된 프로젝트를 Fork 해서 사용

	따라할 일!	뭘 하는건가요?
1	깃허브 저장소 Fork	지킬 테마 깃허브 저장소를 포크합니다. Minima 테마를 사용하겠습니다.
2	저장소 이름 변경	Settings에서 저장소 이름을 <사용자명.github.io> 으로 변경해줍니다. 저장소 이름을 위와 같이 설정해주셔야 https://username.github.io 주소에 페이지를 만들어줄 수 있습니다.
3	$ git clone <repository> $ cd <yourDirectory>	깃허브에 생성된 저장소를 내 컴퓨터에 복제합니다. Clone or Download 에서 저장소 주소를 복사해주세요. 복제한 후에는 해당 디렉토리로 이동해 줍니다. 디렉토리 이름은 저장소 이름과 같습니다.
4	config.yml 파일 편집, 저장 _posts 에 마크다운 생성	config.yml 파일에서 사용자 이름, 연락처, 블로그 제목 등을 알맞게 수정해줍니다. _posts 디렉토리는 실제로 게시물을 보관하는 디렉토리입니다. 게시물 파일 이름은 YYYY-MM-DD-제목.md 로 맞춰주세요.
5	$ git add *	로컬에서 발생한 변경사항을 모두 스테이징합니다.
6	$ git commit –m “<message>”	스테이징된 변경사항을 모두 적용합니다. 로컬에서 실질적인 업데이트가 이루어지는 파트라고 생각하시면 됩니다.
7	$ git push	로컬의 업데이트 사항을 깃허브에 반영합니다.

질문

목차

DataScience
Lab

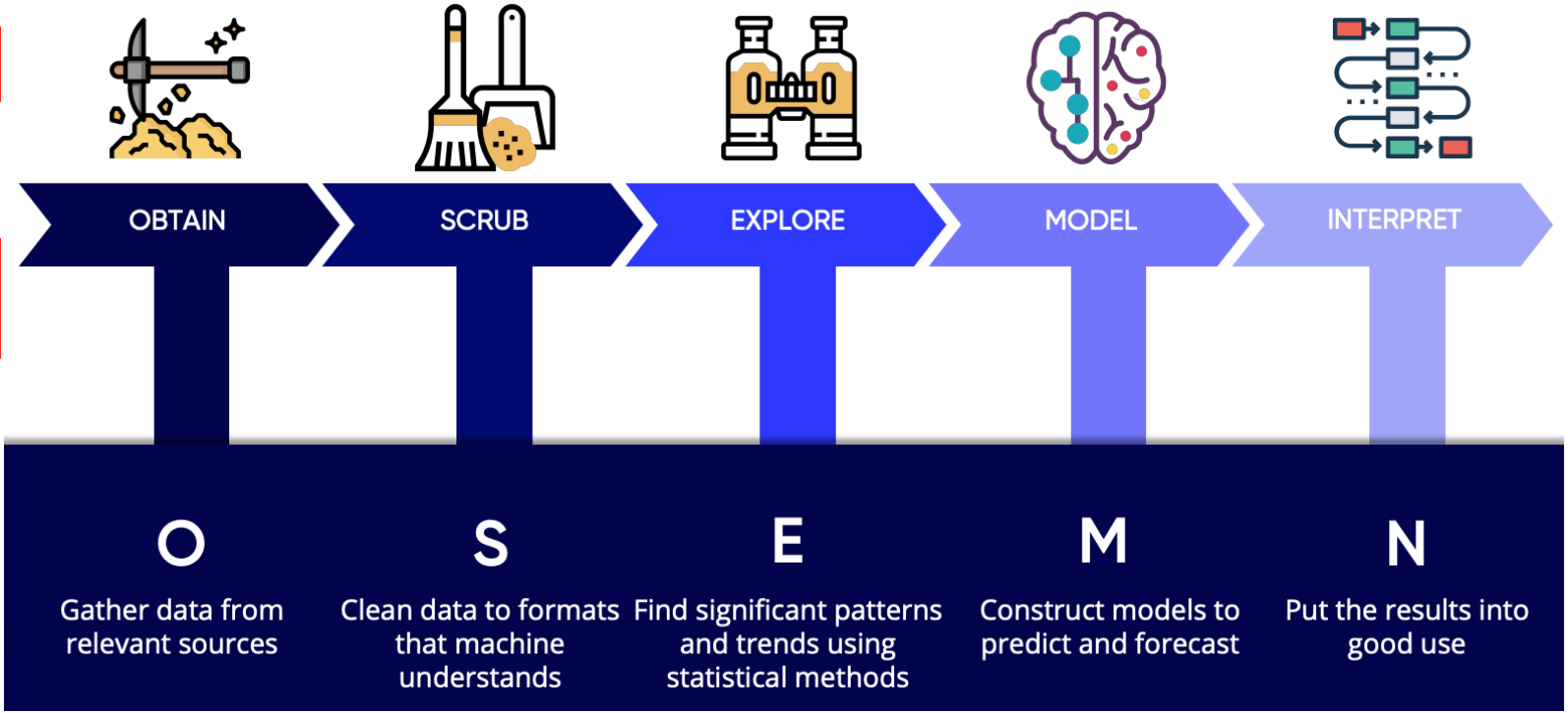
1. EDA 이해하기

EDA in Data Science Process

- 니즈 파악, 발굴
- 결정권자 설득

+ 도메인 지식/ 비즈니스 감각
+ 의사소통

1. PT 능력
2. 시각화

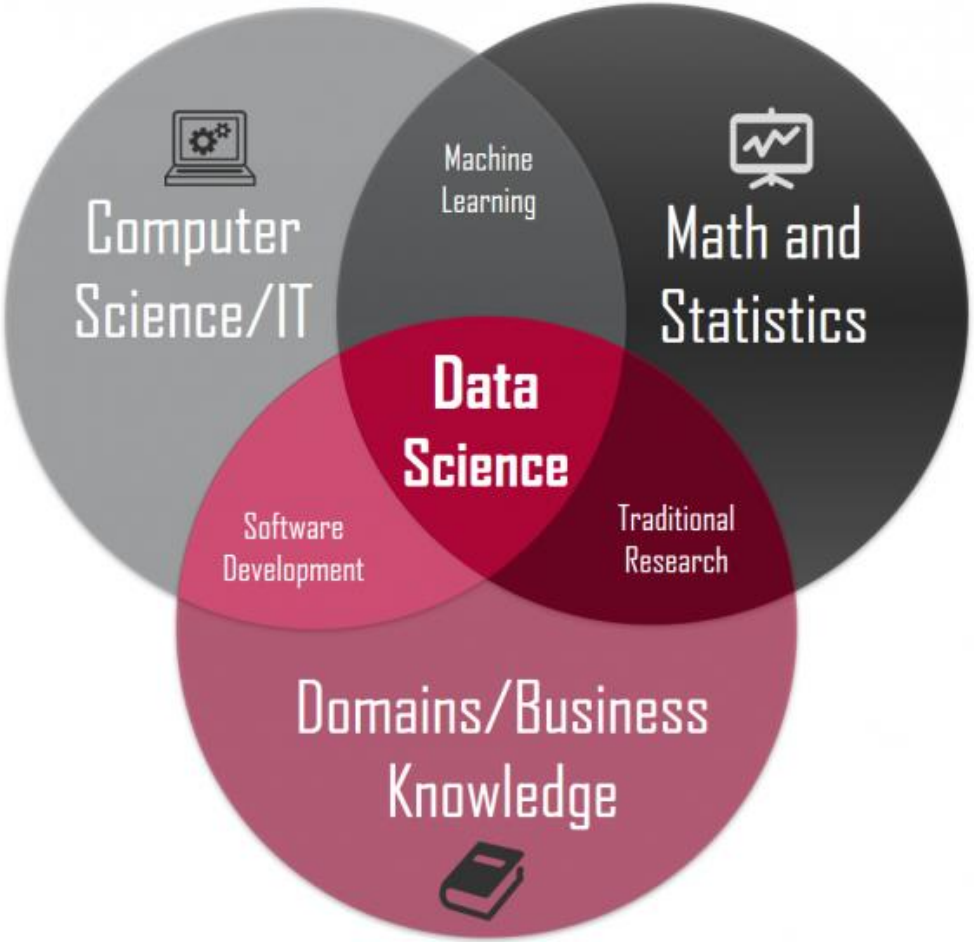


https://www.youtube.com/watch?v=X3paOmcrTjQ&t=14s

EDA in Data Science Process

- 추세
- 패턴
- 외부변수
- 변수간 연관성
- 변수의 특징

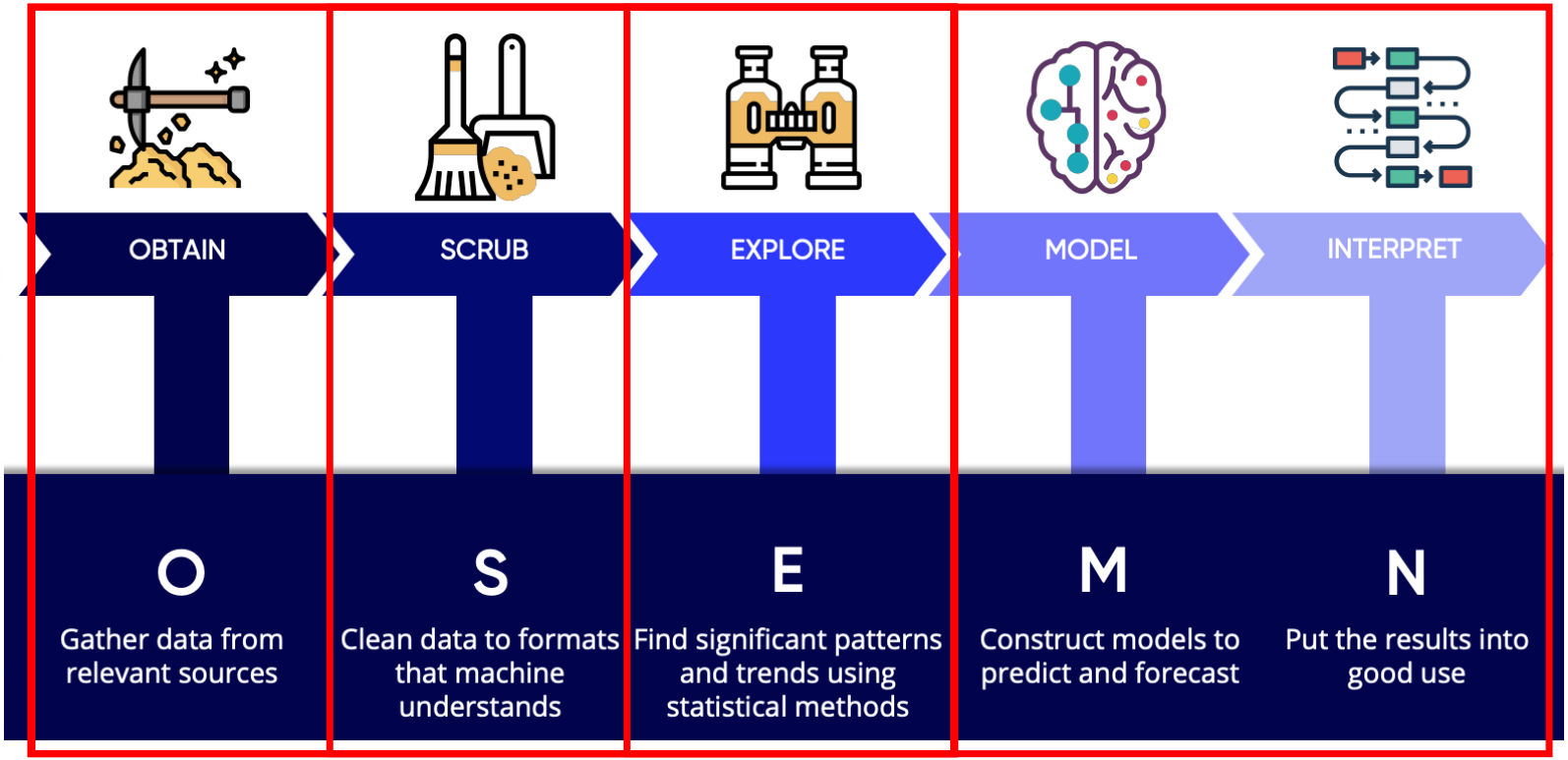
Data : EDA



EDA in Data Science Process

- 추세
- 패턴
- 외부변수
- 변수간 연관성
- 변수의 특징
- 기타 등등

Data에서 확인



Exploratory Data Analysis

- 데이터 분석
- 기존 편견 검증
- 새로운 도메인 지식 획득
- 적합한 변수 선택
- 예측의 정확도와 현상 설명력을 높이는데 결정적 역할

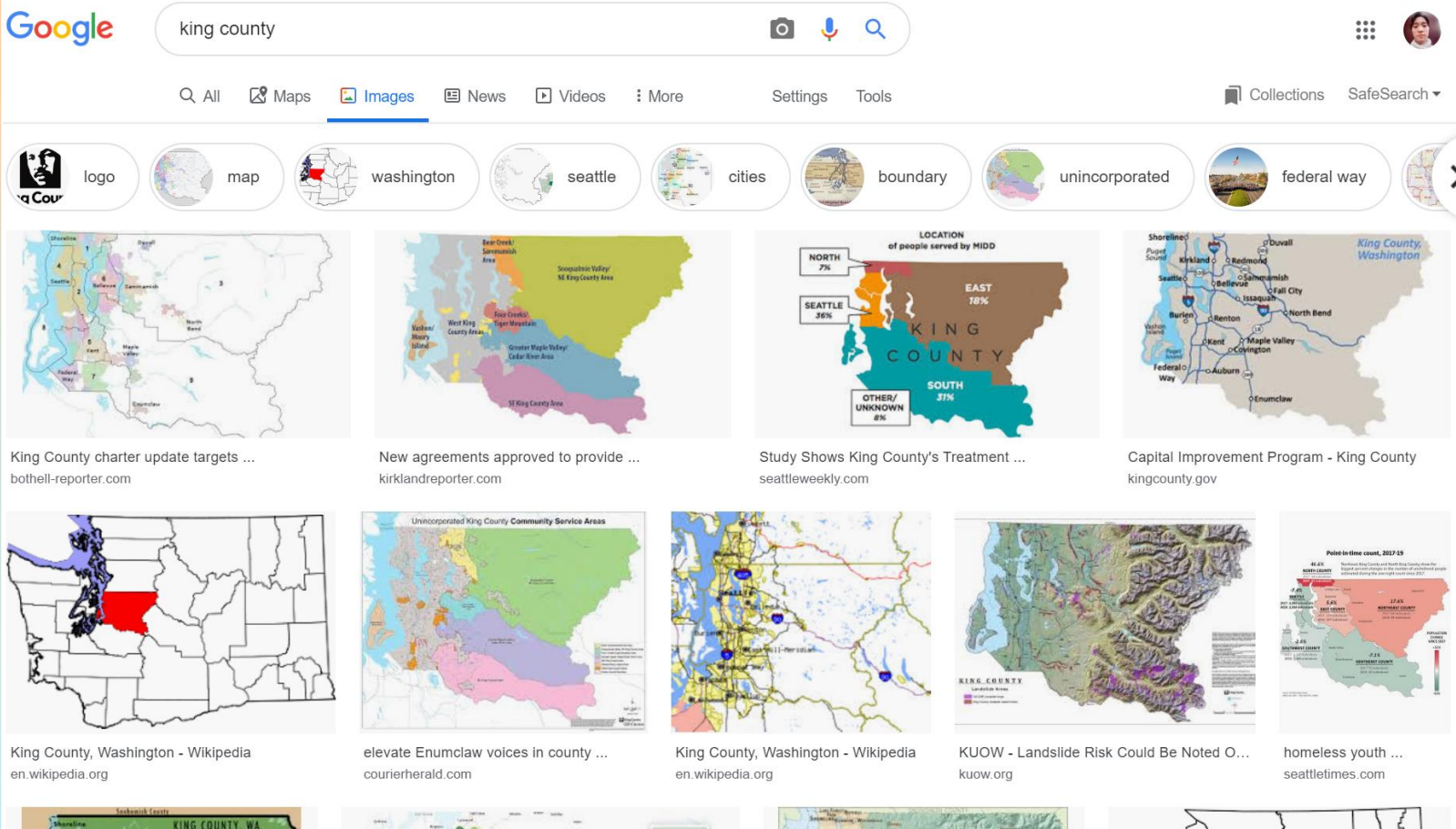
- *좋은 **시나리오**에서 나쁜 **영화**가 나오는 일은 종종 있지만,*
  *나쁜 **시나리오**에서 좋은 **영화**가 나오는 일은 없다*

2. 예시(1): King county

King County in Google

- 지리적 구분
- 경제적 구분
- 사회적 구분
- 사회적 이슈

EDA => 주제



2. 예시(2): audience

DataScience Lab

IGAWorks in Google

- 어떤 회사인지?
- 회사의 최근동향/ 굵직한 사건, 사고
- 회사가 밀고 있는 사업 아이템이 있는지?
- 유사한 자료를 찾을 수 있는지?
- 광고 업계는 어떤 구성원들로 이루어져 있는지?

EDA in audience.csv

	device_ifa	age	gender	marry	install_pack	cate_code	predicted_house_price
0	pNsdu6NgYi	6	M	M	p25485g,p26591g,p32892g,p13661g,p12912g,p16508...	20008:5,21001:1,01003:2,14004:2,06009:2,03003:...	NaN
1	wkArj04TVP	6	F	M	p44316g,p14119g,p16467g,p3022g,p12928g,p10122g...	09001:1,13002:3,01003:1,16004:3,18002:1,21007:...	NaN
2	eEpLI32LWY	7	F	M	p12957g,p17521g,p12912g,p13081g,p31860g,p16620...	16002:5,19001:4,04011:1,p0011:1,18004:3,p0010:...	12900.0

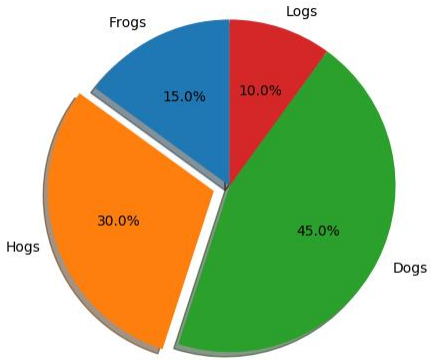
EDA in audience.csv

과제

cate_code의 '문항'에 해당하는 부분을 시작하는 숫자를 기준으로 구분할 때, 문항별 개수를 아래의 방법으로 시각화하세요.

1. (필수) 히스토그램

2. (선택) Pie chart



EDA in audience.csv

유의점

- EDA를 통해 시각화 자료를 만들 때
- 반드시 Python만을 이용할 필요가 없음.

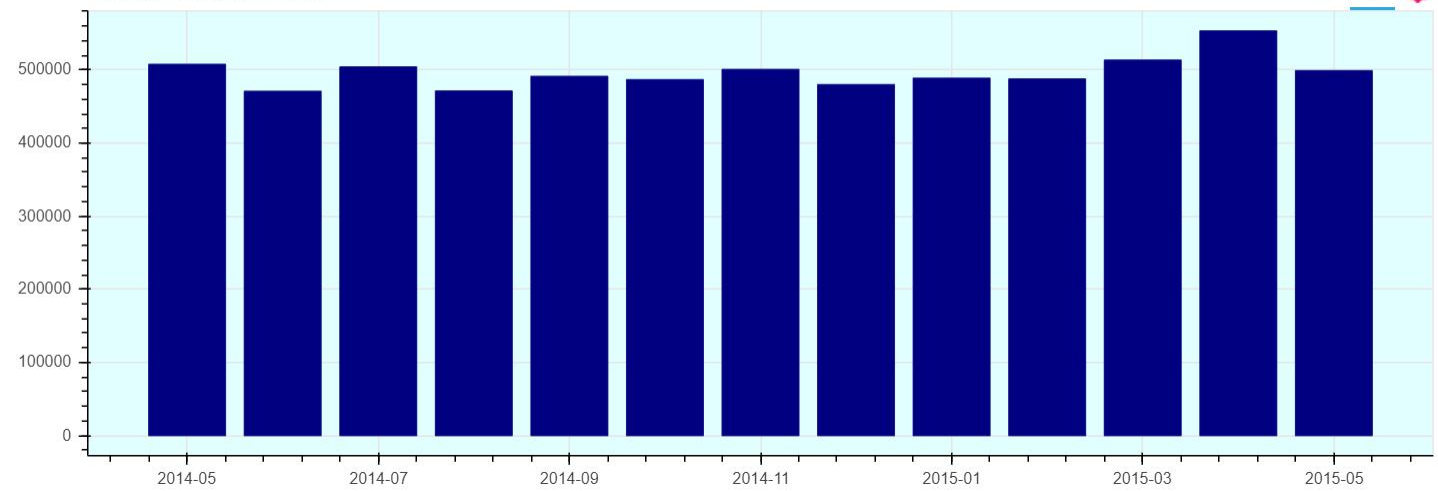
- 물론, matplotlib, bokeh 등의 패키지를 이용할 수 있으나,
- 반드시 해당 언어의 패키지를 쓰는 것만이 정답은 아님.
- 오히려 각각의 영역에 특화된 패키지를 사용하는 것이 유용.
- 따라서 중요한 EDA 데이터는 csv 등으로 저장할 것을 추천

- 그러나 과제의 pie chart는 파이썬 패키지를 사용해볼 것!
- 랜덤으로 한 명 발표
- 기한 : 다음주 화요일

2. 예시(3): terror.csv

3. 시각화

HOUSE AVERAGE PRICE

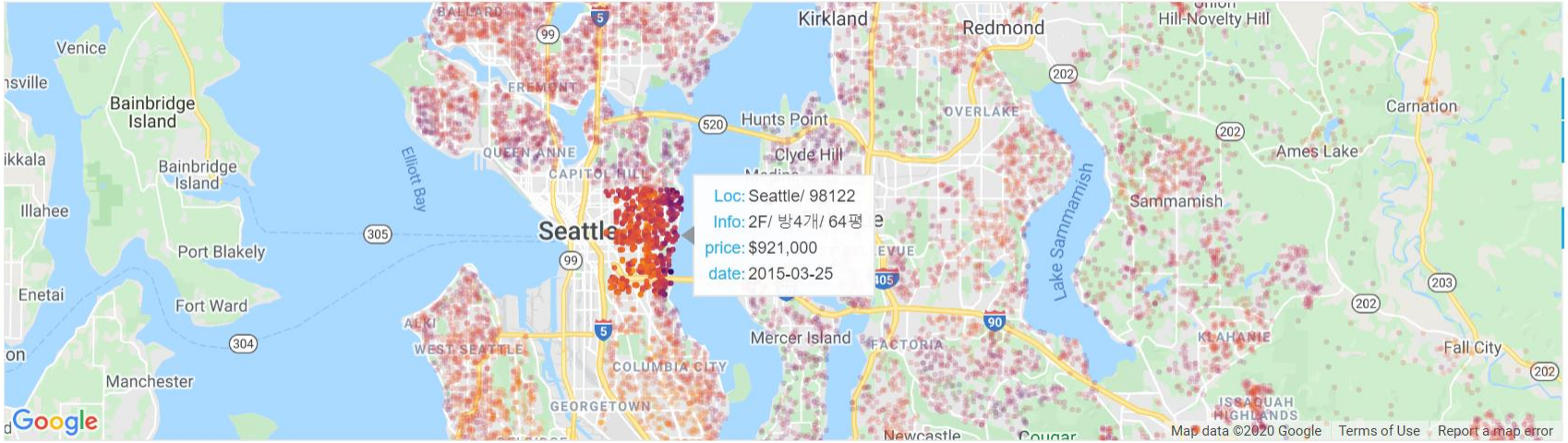


평수: 50

Select Your City

Seattle

Recently sold houses in King County



visualization

한스 로슬링 : https://www.ted.com/talks/hans_rosling_the_best_stats_you_ve_ever_seen

1. Matplotlib
2. Seaborn
3. Bokeh

4. shiny

질문