▣ **강의명:** CSCI E-103: 재현 가능한 머신러닝

▣ **주차:** Lecture 12

▣ **교수명:** Anindita Mahapatra

Eric Gieseke

▣ **목적:** Lecture 12의 핵심 개념 학습

Contents

1 개요 (Overview) **2**

2 용어 정리 (Terminology) **3**

3 핵심 개념과 원리 **4**

3.1 1. 정보 거버넌스 vs 데이터 거버넌스 vs AI 거버넌스 4

3.2 2. 성숙도 모델 (Maturity Models) 4

3.3 3. 거버넌스와 민첩성의 균형 (Trade-off) 4

4 기술적 구현: Databricks Unity Catalog (UC) **5**

4.1 1. 계층 구조 (Hierarchy) 5

4.2 2. 관리형 테이블 vs 외부 테이블 (Managed vs External) 5

4.3 3. Lakehouse Federation (연합) 5

4.4 4. Delta Sharing (공유) 5

5 고급 기능: 보안과 자동화 (Security & Automation) **6**

5.1 1. 데이터 분류 (Data Classification) 6

5.2 2. 속성 기반 접근 제어 (ABAC) 6

5.3 3. 데이터 품질 모니터링 (Data Quality Monitoring) 6

5.4 4. 리니지 (Lineage) 6

6 거버넌스 운영 모델 (Operational Model) **7**

7 자주 묻는 질문 (FAQ) **7**

8 마무리 요약 및 체크리스트 **8**

1 개요 (Overview)

이 문서는 **Data & AI Governance(데이터 및 AI 거버넌스)**의 핵심 이론과 이를 구현하는 기술(Databricks Unity Catalog 등)을 다룹니다.

□ 핵심 요약

핵심 요약

- **목표:** 데이터의 가치를 최대화하면서 동시에 보안 및 규제 리스크를 최소화하는 것.
- **변화:** 과거에는 데이터만 관리했지만, 이제는 AI 모델과 그 결과물까지 관리 범위가 확장됨.
- **도구:** Databricks의 **Unity Catalog**를 중심으로 중앙화된 접근 제어, 감사(Audit), 혈통(Lineage) 추적을 구현.
- **핵심 기능:** 데이터를 자동으로 분류(Tagging)하고, 속성 기반 접근 제어(ABAC)를 통해 민감 정보를 마스킹하며, 외부 데이터베이스까지 통합 관리(Federation)함.

왜 이 내용을 배워야 하나요?
데이터는 '새로운 원유'라고 불릴 만큼 가치가 큽니다. 하지만 관리가 안 된 데이터는 유출 사고 시 기업에 막대한 벌금과 신뢰 하락을 초래합니다. 데이터를 안전하게 지키면서도 필요할 때 바로 꺼내 쓸 수 있게 만드는 '규칙'과 '시스템'을 배우는 것이 이 강의의 목표입니다.

2 용어 정리 (Terminology)

초심자가 혼동하기 쉬운 핵심 용어를 정리했습니다.

Table 1: 거버넌스 핵심 용어 비교표

gray!20 **용어 (한글/영문)**	**쉬운 설명 (비유)**	**기술적 의미**
거버넌스 (Governance)	**교통 법규**: 사고 안 나게 정한 규칙	정책, 표준, 절차를 수립하여 조직을 통제하는 프레임워크
관리 (Management)	**운전 행위**: 법규를 지키며 실제 운전함	거버넌스에서 정한 규칙을 매일 실행하는 운영 활동
메타스토어 (Metastore)	**도서관 목록 카드**: 책 위치 정보 저장소	데이터의 구조(스키마), 위치, 권한 정보를 저장하는 최상위 컨테이너
Unity Catalog (UC)	**통합 신분증**: 어디서든 통하는 ID 카드	Databricks에서 파일, 테이블, 모델 등 모든 자산을 통합 관리하는 계층
ABAC	**꼬리표 검사**: "빨간 딱지 붙은 건 못 봐"	속성(Attribute) 기반 접근 제어. 태그(Tag)를 통해 권한을 자동화함
리니지 (Lineage)	**족보/가계도**: 이 데이터의 조상은 누구?	데이터가 어디서 생성되어 어떻게 변환되고 어디로 흘러갔는지 추적
Federation	**대사관**: 남의 땅에 있지만 우리 법 적용	외부 DB(Snowflake 등)의 데이터를 복사하지 않고 연결하여 조회/관리

3 핵심 개념과 원리

3.1 1. 정보 거버넌스 vs 데이터 거버넌스 vs AI 거버넌스

거버넌스는 범위에 따라 크게 세 가지 층위로 나뉩니다. 가장 큰 우산이 '정보 거버넌스'입니다.

정보 거버넌스 (Information Governance)
조직의 모든 정보(종이 서류, 디지털 파일, 지식 등)를 관리하는 가장 큰 개념입니다.
- 예: "퇴근할 때 책상 위 기밀 서류 치우기", "노트북 잠금 화면 설정하기"

데이터 거버넌스 (Data Governance)
디지털 데이터의 품질, 보안, 수명 주기를 관리합니다.
- 목표: 데이터의 정확성, 일관성, 신뢰성 확보.
- 예: "고객 테이블에 접근할 수 있는 사람은 누구인가?", "이 데이터는 암호화되었는가?"

AI 거버넌스 (AI Governance)
AI 모델의 개발, 배포, 윤리적 사용을 관리합니다. 데이터 거버넌스 없이는 불가능합니다.
- 목표: 편향성(Bias) 방지, 설명 가능성(Explainability), 모델의 투명성.
- 예: "이 AI 모델이 특정 인종에게 불리한 대출 심사를 하지는 않는가?", "학습 데이터에 저작권 위반 데이터가 포함되었는가?"

주의: AI 거버넌스는 데이터 거버넌스 위에 쌓입니다 "걷기도 전에 뛸 수 없다"는 말처럼, 데이터가 엉망인 상태(데이터 거버넌스 부재)에서는 안전하고 공정한 AI(AI 거버넌스)를 만들 수 없습니다.

3.2 2. 성숙도 모델 (Maturity Models)

조직이 거버넌스를 얼마나 잘하고 있는지 5단계로 평가합니다. 단계를 건너뛰는 것(Skip)은 보통 실패합니다. 차근차근 밟아 올라가야 합니다.

1. **초기/인지 (Initial/Aware):** 규칙 없음. 스타트업 초기 단계. 각자 알아서 함.
2. **반응형 (Managed/Reactive):** 문제가 터지면 수습함. 부분적인 규칙 존재.
3. **정의됨 (Defined/Proactive):** 전사적인 표준과 프레임워크가 잡힘. (대부분의 기업 목표)
4. **정량화 (Quantified):** 거버넌스 성과를 수치로 측정 가능함.
5. **최적화 (Optimized):** AI가 자동으로 위반 사항을 감지하고 조치함. 지속적 개선.

3.3 3. 거버넌스와 민첩성의 균형 (Trade-off)

모든 데이터를 금고에 가두면 안전하지만 아무도 일을 못 합니다(민첩성 저하). 반대로 다 풀어주면 빠르지만 위험합니다.

- **고위험 데이터 (개인정보, 금융정보):** 엄격한 통제 (Strict Governance). 접근 절차가 까다로움.
- **저위험 데이터 (공개 데이터, 실험용 데이터):** 느슨한 통제 (Permissive Governance). 혁신을 위해 빠르게 접근 허용.

4 기술적 구현: Databricks Unity Catalog (UC)

이론을 실제 시스템으로 구현하는 도구입니다. Databricks는 **Unity Catalog**라는 단일 계층을 통해 모든 데이터와 AI 자산을 관리합니다.

4.1 1. 계층 구조 (Hierarchy)

UC는 3단계 구조로 데이터를 정리합니다. (파일 시스템의 폴더 구조와 비슷합니다.)

$$\text{Metastore} \xrightarrow{\text{포함}} \text{Catalog} \xrightarrow{\text{포함}} \text{Schema (Database)} \xrightarrow{\text{포함}} \text{Table / Volume / Model}$$

- **Metastore:** 최상위 컨테이너. 보통 리전(Region) 당 하나.
- **Catalog:** 데이터 자산의 가장 큰 그룹. (예: 'prod', 'dev', '$\text{hr}_{d}ata$')
- **Schema:** 테이블과 뷰를 담는 논리적 그룹.
- **Table/Volume:** 실제 데이터. (Table은 정형 데이터, Volume은 비정형 파일)

4.2 2. 관리형 테이블 vs 외부 테이블 (Managed vs External)

- **Managed Table:** Databricks가 데이터 파일의 위치와 수명 주기를 직접 관리합니다. 테이블을 지우면 실제 파일도 지워집니다. (가장 추천됨)
- **External Table:** 데이터 파일은 내 클라우드 스토리지(S3 등)에 있고, Databricks는 그 위치만 참조합니다. 테이블을 지워도 원본 파일은 남습니다.

4.3 3. Lakehouse Federation (연합)

외부 데이터베이스(MySQL, Snowflake, Postgres 등)를 데이터를 복사해오지 않고(No Copy), 마치 로컬 테이블처럼 조회하는 기능입니다.

□ 예제:

비유: 대사관 Databricks 안에 있는 'Snowflake Catalog'는 대사관과 같습니다. 실제 영토(데이터)는 Snowflake에 있지만, Databricks의 법(거버넌스 규칙)을 적용하여 조회할 수 있습니다. 쿼리를 날리면 Databricks가 처리하는 게 아니라 원본 DB(Snowflake)로 쿼리를 보내서 결과만 받습니다(Pushdown).

4.4 4. Delta Sharing (공유)

서로 다른 플랫폼 간에 데이터를 안전하게 공유하는 개방형 프로토콜입니다.

- 데이터를 복제해서 이메일로 보내거나 FTP로 전송할 필요가 없습니다.
- 받는 사람이 Databricks를 안 써도 됩니다(Pandas, Tableau, PowerBI 등으로 직접 접속 가능).

5 고급 기능: 보안과 자동화 (Security & Automation)

5.1 1. 데이터 분류 (Data Classification)

시스템이 자동으로 데이터를 스캔하여 이메일, 신용카드 번호 같은 **민감 정보(PII)**를 찾아냅니다. 찾으면 자동으로 태그(Tag)를 붙입니다.

5.2 2. 속성 기반 접근 제어 (ABAC)

과거에는 "철수에게 A 테이블 권한 주기" 식(RBAC)이었다면, 이제는 "'기밀' 태그가 붙은 건 관리자만 보기" 식으로 규칙을 만듭니다.

- **Row Level Filtering:** 특정 조건을 만족하는 행(Row)만 보여줌. (예: 내 부서 데이터만 보기)
- **Column Level Masking:** 특정 열(Column)의 데이터를 가림. (예: 주민번호 뒷자리 별표 처리)

```
-- 1. 마스킹함수정의관리자가 ( 아니면별표로표시 )
CREATE FUNCTION ssn_mask(ssn STRING)
RETURN IF(
    is_account_group_member('admin'), -- 관리자그룹인지확인
    ssn,                              -- 관리자면원본노출
    '****'                            -- 아니면별표표시
);

-- 2. 테이블의특정컬럼에마스킹함수적용
ALTER TABLE users
ALTER COLUMN social_security_number
SET MASK ssn_mask;
```

Listing 1: SQL을 이용한 마스킹 정책 생성 예시

5.3 3. 데이터 품질 모니터링 (Data Quality Monitoring)

데이터가 썩지 않았는지 감시합니다.

- **Freshness (최신성):** 데이터가 제때 들어왔는가?
- **Completeness (완전성):** 데이터 양이 갑자기 줄지 않았는가? (어제는 1000행이었는데 오늘은 0행?)
- 별도 설정 없이 버튼 클릭 한 번으로 자동 감시가 가능합니다.

5.4 4. 리니지 (Lineage)

데이터의 가계도입니다. "이 차트의 숫자가 왜 이상하지?"라는 질문이 나왔을 때, 그 데이터가 어떤 원천 테이블에서 와서 어떤 변환 과정을 거쳤는지 시각적으로 보여줍니다. (Source → Transform → Dashboard)

6 거버넌스 운영 모델 (Operational Model)

조직의 크기와 문화에 따라 거버넌스를 누가 주도할지 결정해야 합니다. 정답은 없으며, 조직 상황에 맞춰 선택합니다.

Table 2: 거버넌스 운영 모델 비교

gray!20 **모델**	**특징**	**장점**	**단점/적합 조직**
중앙 집중형	중앙 팀이 모든 규칙 결정	일관성 높음, 보안 강력	느림, 병목 현상 발생 (규제 산업)
분산형	각 부서가 알아서 관리	혁신 속도 빠름, 유연함	표준 없음, 중복 발생 (스타트업)
연합형 (Federated)	중앙 가이드 + 부서 실행	자율성과 통제의 균형	조율이 어려울 수 있음 (대기업)
하이브리드	핵심 데이터는 중앙, 나머지는 분산	중요 정보 보호 + 업무 효율	구조가 복잡함

7 자주 묻는 질문 (FAQ)

Q. 클라우드(AWS/Azure)의 IAM 역할만 쓰면 안 되나요? 왜 Unity Catalog가 필요한가요?
A. 클라우드 IAM은 '파일'이나 '폴더' 단위의 접근은 제어할 수 있지만, '테이블의 특정 컬럼'이나 '행' 단위의 정교한 제어는 어렵습니다. UC는 데이터 내용에 기반한 세밀한(Fine-grained) 제어를 가능하게 합니다.

Q. 데이터 거버넌스를 도입하면 업무 속도가 느려지지 않나요?
A. 초기에는 규칙 설정 때문에 느려 보일 수 있습니다. 하지만 장기적으로는 '데이터를 찾는 시간', '데이터 품질을 의심하고 검증하는 시간', '보안 사고 수습 시간'을 줄여주어 전체적인 속도는 빨라집니다.

Q. ABAC과 RBAC의 차이는 무엇인가요?
A. **RBAC(Role-Based)**은 "팀장님은 다 볼 수 있어"처럼 역할에 권한을 줍니다. **ABAC(Attribute-Based)**은 "누구든 '기밀' 태그가 붙은 건 못 봐"처럼 데이터의 속성(태그)에 따라 동적으로 권한을 줍니다. ABAC이 더 유연하고 확장성이 좋습니다.

8 마무리 요약 및 체크리스트

이 문서를 통해 학습한 내용을 점검해 보세요.

학습 체크리스트

- ☐ 정보, 데이터, AI 거버넌스의 포함 관계를 이해했는가?
- ☐ 성숙도 모델 5단계를 순서대로 나열할 수 있는가?
- ☐ Unity Catalog의 3계층 구조(Metastore-Catalog-Schema)를 그릴 수 있는가?
- ☐ ABAC의 개념과 왜 태깅(Tagging)이 중요한지 설명할 수 있는가?
- ☐ Lakehouse Federation이 데이터를 복사하지 않고(No Copy) 조회한다는 의미를 아는가?
- ☐ 데이터 품질 모니터링의 두 가지 핵심 지표(Freshness, Completeness)를 이해했는가?

□ 핵심 요약

1페이지 요약 **1. 거버넌스는 전략이다:** 단순한 제약이 아니라, 데이터를 자산으로 만들기 위한 필수 전략입니다.
2. 통합 관리: Unity Catalog를 통해 파일, 테이블, 모델을 한 곳에서 관리합니다.
3. 자동화: AI를 활용해 민감 정보를 자동 분류하고, 품질을 모니터링합니다.
4. 연결성: Federation과 Sharing을 통해 데이터 사일로(고립)를 제거하고 외부와 연결합니다.
5. 균형: 보안(Security)과 민첩성(Agility) 사이에서 조직에 맞는 적절한 운영 모델을 선택해야 합니다.