



Google BigQuery와 Tableau: 모범 사례

소개

Tableau와 Google BigQuery는 사람들이 사용하기 쉬운 시각적 인터페이스를 통해 대규모의 데이터를 신속하게 분석하고 질문에 대한 답을 구할 수 있도록 지원합니다. 두 가지 도구를 함께 사용하면 다음과 같은 이점을 누릴 수 있습니다.

- Google BigQuery의 강력한 기능을 빠른 대화형 분석 작업을 수행할 수 있도록 일반 사용자에게 제공합니다.
- 코드 작성이나 서버 관리가 전혀 필요하지 않으며, 시각적 분석 도구를 사용해 수십억 개의 행을 몇 초 내에 분석합니다.
- Google BigQuery 데이터에 연결하여 멋진 대시보드를 몇 분 내에 만들고 이를 통해 신속하게 최신 자료를 활용합니다.
- Tableau Server 및 Tableau Online을 통해 웹에서 보고서와 인사이트를 공유하고 누구든지 모든 기기에서 연결할 수 있습니다.
- Google BigQuery의 클라우드 대응력을 Tableau의 빠른 속도와 결합하여 프로젝트의 가치를 더욱 신속하게 실현합니다.

두 기술을 함께 사용하면 성능이 크게 향상되어 디자인 주기가 단축되고 사용자와 조직이 더욱 효율적으로 업무를 수행할 수 있습니다. 이 백서에서는 비주얼리제이션의 응답 속도를 극대화하기 위해 데이터 모델링과 쿼리 생성을 최적화하는 기술을 설명합니다. 또한 Tableau와 BigQuery를 함께 사용할 때 최고의 비용 효율성을 달성할 수 있는 기술에 대해서도 설명합니다.

작성자

Pierce Young, Tableau 제품 관리자

Vaidy Krishnan, Tableau 수석 제품 관리자

Riley Maris, Tableau 제품 마케팅 수석 전문가

Babu Prasad Elumala, Google 솔루션 엔지니어

Seth Hollyman, Google 기술 프로그램 관리자

Tino Tereshko, Google 엔터프라이즈 솔루션 엔지니어

Mike Graboski, Google 솔루션 엔지니어

목차

기술 개요	4
Google BigQuery	4
Tableau	5
성능을 위한 모범 사례: Tableau	6
성능 기록기	6
컨텍스트 필터	7
집합 및 그룹	8
필터 먼저 추가하기	8
자동 업데이트 해제	9
경고 찾기	9
병렬 쿼리 최적화하기	10
비용 및 성능을 위한 모범 사례: Google BigQuery	10
비정규화 및 사전 조인	10
날짜 기준으로 테이블 분할	11
많은 유사한 쿼리를 실행하는 경우 대상 테이블 지정하기	12
Tableau를 사용하여 Google BigQuery ML 모델 결과 시각화하기	12
사례 연구: Tableau와 Google BigQuery로	
셀프 서비스 분석을 구축한 zulily의 유용한 팁	13
결론	13
Tableau 정보	14
추가 리소스	14

기술 개요

Google BigQuery

BigQuery는 일반 SQL에서 몇 초 내에 페타바이트 규모의 데이터를 처리할 수 있으며 세부적인 조정이나 특별한 기술이 필요하지 않습니다. 대규모의 데이터 집합을 분석하기 위한 Google의 혁신적인 기술인 Dremel을 기반으로 하는 BigQuery는 대규모 기업이 이전에 수백만 달러의 비용을 들여야 얻을 수 있는 성능 수준을 기가바이트당 매우 저렴한 가격으로 제공합니다.

BigQuery는 대규모의 정형 및 반정형 데이터 집합을 대상으로 SQL 쿼리를 실행하는 데 가장 적합한 데이터 웨어하우스입니다. 사용 사례 및 데이터 집합의 예는 다음과 같습니다.

- 애드혹 분석
- 웹 로그
- 컴퓨터/서버 로그
- 사물 인터넷 데이터 집합
- 전자상거래 고객 행동
- 모바일 앱 데이터
- 소매업 분석
- 게임 원격 측정
- Google Analytics Premium 데이터
- 기존 RDBMS에서 일괄 쿼리를 실행하는 데 수 분(또는 수 시간)이 걸리던 모든 데이터 집합

BigQuery는 운영 및 유지 관리가 필요하지 않으며 Google Cloud Platform과 통합됩니다. 다른 클라우드 기반 분석 솔루션과는 달리, BigQuery를 사용하면 서버 클러스터를 미리 프로비저닝할 필요가 없습니다. 프로세싱 클러스터는 런타임 시 BigQuery에서 크기를 지정하고 프로비저닝합니다.

데이터의 크기가 증가함에 따라 BigQuery는 자동으로 프로세싱 파워를 추가하지만 기가바이트당 지불하는 비용은 동일합니다.

기존 SQL 및 표준 SQL

Google BigQuery는 BigQuery SQL(기존 SQL이라고 함) 이외에 표준 SQL도 사용하도록 API를 업그레이드하였고, Tableau는 이러한 변화를 지원하기 위해 Google BigQuery 커넥터를 업그레이드했습니다. BigQuery 사용자가 표준 SQL을 사용하면 세부 수준 식, 메타데이터 유효성 검증 속도 향상, 연결 대금 청구 프로젝트 선택 기능 등 여러 이점을 얻을 수 있습니다. 이 가이드는 표준 SQL 사용을 전제로 작성되었습니다.

기존 SQL에서 표준 SQL로의 마이그레이션에 대한 자세한 내용은 [이전 SQL에서 이전에 대한 온라인 도움말 가이드](#)를 참조하십시오.

Tableau

Tableau는 사람들이 데이터를 보고 이해할 수 있도록 지원합니다. 스탠포드 대학에서 개발된 기술을 기반으로 하는 Tableau의 최신 분석 플랫폼은 평범한 사람들에게도 데이터의 힘을 실어줍니다. 이를 통해 광범위한 범위의 많은 사용자가 데이터를 활용하여 질문하고, 문제를 해결하고, 인사이트를 공유하고, 가치를 창출하고 있습니다. 사용자는 기존 BI 도구에 익숙한 지 여부에 관계없이 직관적인 끌어 놓기 사용자 인터페이스를 통해 Tableau를 활용하여 풍부한 대화형 비주얼리제이션 및 강력한 대시보드를 작성하고 탐색하는 방법을 빠르게 습득합니다

최근에는 Tableau 플랫폼의 기능을 확장하여, 시각적이고 직접적인 스마트 데이터 준비 기능과 자연어로 게시된 데이터 원본을 쿼리하는 기능을 포함하였습니다.

Tableau 기본 제공 최적화 기능

데이터 원본 커넥터: Tableau는 라이브 데이터 연결 및 인메모리 추출을 모두 지원하는 Google BigQuery에 최적화된 커넥터를 내장하고 있습니다. Tableau의 데이터 통합 기능을 통해 사용자는 67개의 지원되는 다른 데이터 원본 데이터를 BigQuery 데이터와 통합할 수 있습니다. Tableau Server 또는 Tableau Online을 통해 클라우드에 게시된 비주얼리제이션의 경우 Google BigQuery에 대한 직접 연결을 유지할 수 있습니다.

병렬 쿼리: Tableau는 Google BigQuery와 기타 데이터 원본의 기능을 활용하여 최대 16개의 다중 쿼리를 동시에 실행합니다. 결과가 아직 캐시되지 않은 경우 독립 쿼리 및 중복 제거 쿼리의 일괄 작업은 그룹화되어 BigQuery로 전송됩니다. BigQuery의 확장 아키텍처를 통한 병렬 쿼리 덕분에 사용자는 성능이 크게 향상되는 것을 경험하게 됩니다.

쿼리 융합: Tableau는 가능한 경우 통합 문서 및 대시보드의 여러 쿼리를 융합함으로써 BigQuery로 전송되는 쿼리 수를 줄입니다. Tableau는 먼저, 반환되는 열에서 다른 부분을 제외하고 유사한 쿼리를 식별합니다. 그런 다음 집계 또는 사용자 계산 수준만 다른 경우 쿼리를 결합합니다.

외부 쿼리 캐시: 동일한 쿼리를 마지막으로 실행한 후 기초 데이터 원본이 변경되지 않은 경우 Tableau는 자동으로 이전에 저장된 쿼리 캐시를 읽어 거의 즉각적으로 로드합니다.

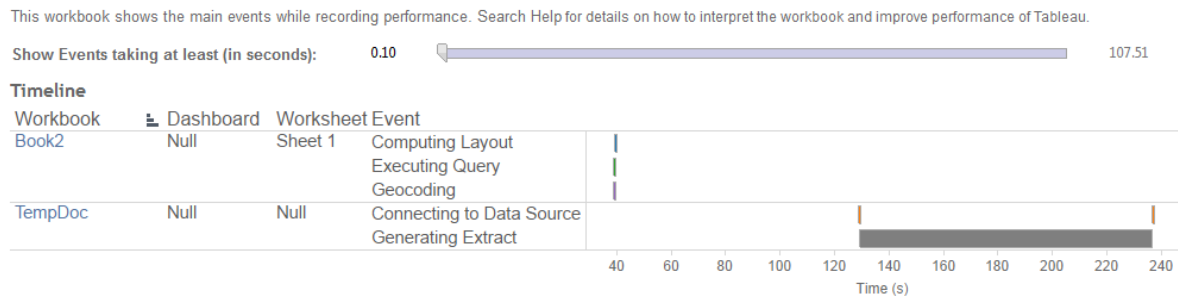
Tableau Desktop에서 온디맨드 연결: 사용자가 게시된 통합 문서를 열면, Tableau Desktop은 현재 시트의 데이터를 표시하는 데 필요한 데이터 원본에만 연결합니다. 즉, 데이터를 더 빠르게 확인할 수 있습니다.

성능을 위한 모범 사례: Tableau

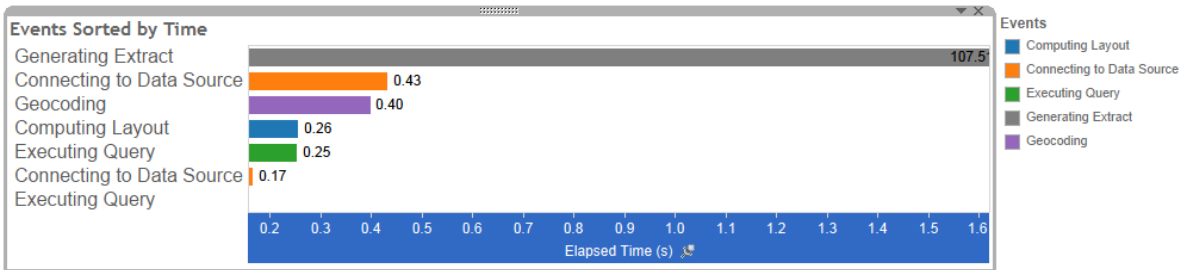
추가 도구 및 사용자 지정 설정을 시작하기 전에 가능한 한 Tableau 배포를 계속 업데이트하시는 것이 좋습니다. 이렇게 하면 Tableau에서 계속 발표하는 최신 버전의 향상된 성능을 활용할 수 있습니다.

성능 기록기

기본 제공되는 강력한 도구인 성능 기록기를 사용하면 느린 쿼리를 정확하게 찾고 통합 문서를 최적화하여 성능을 극대화할 수 있습니다. 성능 기록기는 개별 통합 문서가 쿼리를 실행하고 레이아웃을 계산할 때까지 경과한 시간을 추적함으로써 이 작업을 수행합니다. 아래의 녹색 막대 위로 마우스를 이동하면 BigQuery에서 생성되는 쿼리가 사용자에게 표시됩니다. 느린 쿼리를 파악한 후 데이터 모델로 다시 돌아가 성능 문제를 해결할 수도 있습니다.



일정 뷰에서 통합 문서, 대시보드 및 워크시트 열은 이벤트에 대한 컨텍스트를 식별합니다.



기간이 긴 이벤트를 사용하면 통합 문서의 속도를 향상하기 위해 가장 먼저 확인할 위치를 알 수 있습니다.

성능 기록 만들기 또는 해석에 대한 자세한 내용은 아래를 참조하십시오.

[Tableau Desktop에서 성능 기록 만들기](#)

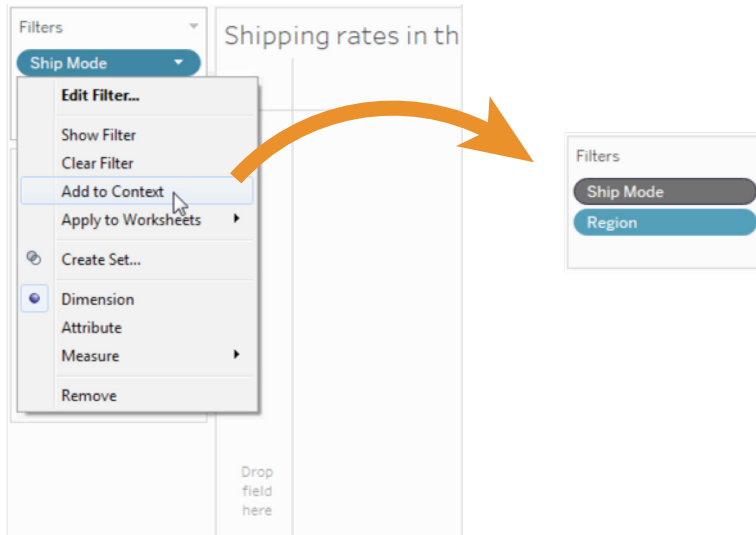
[Tableau Server에서 성능 기록 해석](#)

컨텍스트 필터

규모가 큰 데이터 원본에 필터를 적용하는 경우 컨텍스트 필터를 설정하면 성능을 개선할 수 있습니다. 컨텍스트 필터는 데이터 원본에 먼저 적용되므로 추가 필터는 결과 레코드에만 적용됩니다. 이 절차로 인해 데이터 원본의 각 레코드에 각 필터를 적용하지 않아도 됩니다.

데이터 집합의 크기를 현저히 줄이는 필터를 설정하고 많은 데이터 뷰에 해당 필터를 사용하려는 경우에는 해당 필터를 컨텍스트 필터로 설정하는 것이 좋습니다.

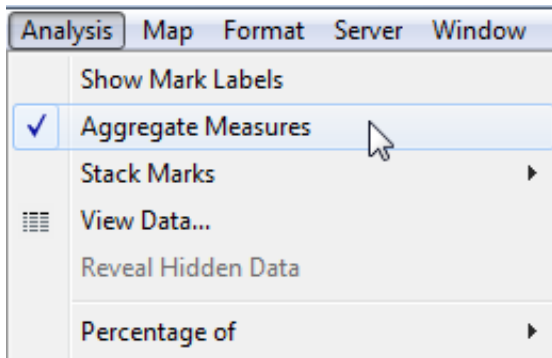
자세한 내용은 [컨텍스트 필터로 뷰 성능 향상에 대한 온라인 도움말 가이드](#)를 참조하십시오.



한 개 이상의 컨텍스트 필터를 설정하여 성능을 향상할 수 있습니다.

측정값 집계

만든 뷰가 느린 경우, 집계되지 않은 측정값이 아니라 이미 집계된 측정값으로 작업하고 있는지 확인해야 합니다. 뷰가 느린 경우 일반적으로 한 번에 너무 많은 데이터 행을 보려고 한다는 것을 의미합니다. 데이터를 집계하면 행의 수를 줄일 수 있습니다.



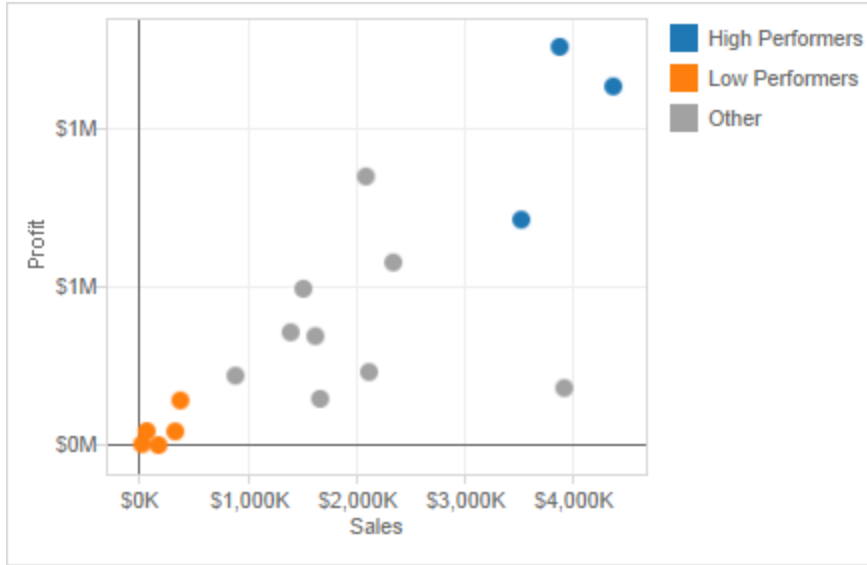
분석 메뉴에서 측정값이 집계되는지 확인하십시오. 측정값에 대한 기본 집계를 설정할 수도 있습니다.

자세한 내용은 [데이터 집계에 대한 온라인 도움말 가이드](#)를 참조하십시오.

집합 및 그룹

측정값 범위를 기준으로 멤버를 제거하기 위해 차원을 필터링하려는 경우 수량 필터를 사용하는 것보다 집합을 만드는 것이 좋습니다. 예를 들어 차원의 모든 항목이 아니라 상위 50개 항목만 반환하는 집합을 만들 수 있습니다.

선택 내용에서 그룹을 만드는 경우 원하는 열만 포함해야 합니다. 집합에 열이 추가될 때마다 성능이 저하됩니다.



Tableau에서 그룹을 만들 때, 남아있는 모든 멤버를 그룹화하는 옵션이 있습니다.

자세한 내용은 [집합 만들기에 대한 온라인 도움말 가이드](#) 및 [그룹 만들기에 대한 온라인 도움말 가이드](#)를 참조하십시오.

필터 먼저 추가하기

대규모 데이터 원본으로 작업을 수행하는 중에 자동 업데이트 기능이 꺼져 있는 경우 뷰에 필터를 추가하면 쿼리가 매우 느려질 수 있습니다. 뷰를 만든 후 필터를 지정하지 말고 먼저 필터를 지정한 다음 뷰에 필드를 끌어 놓는 것이 좋습니다. 이렇게 하면 업데이트를 실행하거나 자동 업데이트 기능을 켤 때 필터가 먼저 적용됩니다.

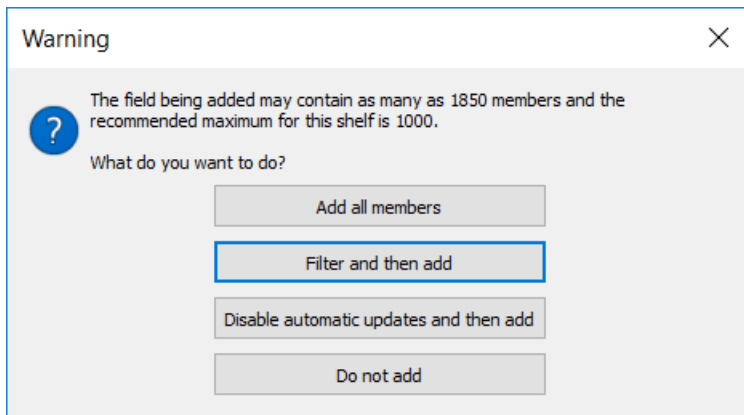
자동 업데이트 해제

선반에 필드를 배치할 때 Tableau는 데이터 원본을 자동으로 쿼리하여 뷰를 생성합니다. 조밀한 데이터 뷰를 만드는 경우 쿼리 시간이 오래 걸릴 수 있으며 시스템 성능이 크게 저하될 수 있습니다. 이 경우에는 뷰를 작성하는 동안 쿼리를 하지 않도록 Tableau에 설정할 수 있습니다. 결과를 확인할 준비가 되면 쿼리를 다시 켤 수 있습니다.

자세한 내용은 [자동 업데이트와 성능에 대한 온라인 도움말 가이드](#)를 참조하십시오.

경고 찾기

선반에 규모가 큰 차원(멤버가 많은 차원)을 배치하려고 시도하면 성능에 대한 경고 대화 상자가 Tableau에 표시됩니다. 대화 상자에는 아래 그림에 표시된 것과 같이 4가지 선택 사항이 제공됩니다. 모든 멤버를 추가하도록 선택하는 경우 성능이 크게 저하될 수 있습니다.



선반에 규모가 큰 차원이 배치되어 성능이 위험한 상태가 되면 Tableau가 경고를 표시합니다.

병렬 쿼리 최적화하기

병렬 쿼리를 구성함으로써, 사용자 지정 특성을 사용하여 BigQuery에서 Tableau Online 및 Tableau Server로 반환되는, 그리고 Tableau Desktop에서 대규모 결과 집합의 성능을 향상할 수 있습니다. 이러한 사용자 지정 특성을 게시된 통합 문서나 데이터 원본에 포함하려면, Tableau Online이나 Tableau Server에 통합 문서나 데이터 원본을 게시하기 전에 이러한 특성을 지정해야 합니다.

자세한 내용은 [Google BigQuery에 대한 온라인 도움말 가이드](#)의 '사용자 지정 특성을 사용하여 쿼리 성능 향상'을 참조하십시오.

비용 및 성능을 위한 모범 사례: Google BigQuery

고성능 쿼리를 수행하고 비용을 절감하려면 일반적으로 Google Cloud Storage와 같이 외부 데이터 원본에 데이터가 결합된 테이블은 사용하지 않는 것이 가장 좋습니다. 그러한 경우 데이터 집합에 대한 반복 쿼리를 수행하려면, BigQuery에서(Tableau가 아니라) Query API를 사용하여 데이터를 구현해야 Tableau에서 데이터 집합에 대한 고성능 쿼리가 가능해집니다.

비정규화 및 사전 조인

BigQuery는 대규모의 조인을 지원하며 조인 성능이 우수합니다. 또한 BigQuery는 열 형식의 데이터 저장소이며 비정규화 데이터 집합을 사용할 때 성능이 극대화됩니다.

클라우드의 한 가지 이점은 저장소를 분리하고 리소스를 계산하여 사용자가 규모를 조정하고 각각의 비용을 따로 지불할 수 있다는 것입니다. BigQuery 저장소는 비용이 저렴하고 확장성이 뛰어나므로 데이터 집합을 비정규화하고 동일한 테이블로 사전 조인하는 것이 좋습니다. 본질적으로 이는 컴퓨팅 리소스는 더 적게, 저장소 리소스는 더 많이 사용함을 의미합니다(후자가 성능 및 비용 면에서 더 효율적임). BigQuery는 열 형식의 저장소로 데이터를 잘 압축할 수 있으므로, 컴퓨팅 리소스를 줄이고 저장소를 늘리는 것은 비용도 절감됩니다.

BigQuery는 우수한 ETL 도구이며, 이를 사용하면 대규모의 변환과 관련 작업을 빠르고 효율적으로 수행할 수 있습니다. 128MB보다 큰 데이터 집합을 구현하는 경우 '대규모 결과 허용' 기능을 사용해야 합니다.

BigQuery의 SQL 언어를 사용하여 로드할 데이터를 준비하는 방법과 데이터를 쿼리하는 방법에 대한 자세한 내용은 다음을 참조하십시오.

[비정규화, 중첩, 반복 데이터 로드
크기가 큰 쿼리 결과 쓰기](#)

날짜 기준으로 테이블 분할

테이블을 더 작은 파티션으로 나누는 것, 즉 '분할'을 하면, 데이터 관리를 단순화하고 쿼리 성능과 비용 측면을 향상할 수 있습니다. 또한 BigQuery는 분할된 테이블에 대한 클러스터 분석을 지원하므로 데이터가 이미 날짜 또는 타임스탬프 열에 분할되어 있거나 쿼리의 특정 열에 대한 필터 또는 집계기가 있는 경우에 유용합니다.

일부 데이터는 날짜별로 자연스럽게 구분됩니다. 예를 들어 로그 데이터 또는 단순히 증가하는 타임스탬프가 포함된 레코드가 있는 모든 데이터가 여기에 해당됩니다. 이 경우 BigQuery 테이블을 날짜 기준으로 분할하고 테이블 이름에 날짜를 포함합니다. 이러한 기능을 사용하려면 Tableau에서 사용자 지정 SQL을 활용해야 합니다.

자세한 내용은 [사용자 지정 SQL 쿼리에 연결에 대한 온라인 도움말 가이드](#)를 참조하십시오.

예를 들어 테이블 이름을 `mytable_20170501`, `mytable_20170502` 등으로 지정해 보십시오.

그런 다음, 날짜 기준으로 필터링하는 쿼리를 실행하려는 경우 다음과 같이 BigQuery의 와일드카드 테이블 함수를 사용할 수 있습니다.

```
SELECT
  name
FROM
  `myProject.myDataSet.mytable_*`
WHERE
  age >= 35
```

위 예제에서, 자동으로 `mytable_` 접두사가 있는 모든 테이블을 포함하게 됩니다.

와일드카드를 사용하려면 '[임의 접두사]YYYYMMDD'와 같은 패턴으로 테이블의 이름을 지정해야 합니다.

다른 데이터베이스 시스템은 성능을 개선하기 위해 분할 기능을 사용합니다. 날짜를 기준으로 분할하는 경우 BigQuery의 성능 차이는 무시해도 되는 정도지만 여기서 중요한 것은 비용입니다. 더 적은 양의 데이터를 처리하면 쿼리당 비용이 적게 들기 때문입니다.

분 단위 수준으로 분할하도록 선택하면 분할이 너무 많아 성능에 직접적으로 영향을 줄 수 있음에 유의하시기 바랍니다. 동시에 너무 많은 분할이 이루어지지 않도록 주의해야 합니다. 일반적으로 일 단위 이상의 수준이 적합합니다.

분할에 대한 자세한 내용은 다음을 참조하십시오.

[파티션 테이블 소개](#)

[클러스터링된 테이블 소개](#)

[와일드 카드 테이블을 사용하여 여러 테이블 쿼리](#)

많은 유사한 쿼리를 실행하는 경우 대상 테이블 지정하기

동일한 쿼리를 여러 개 실행하는 경우 쿼리 캐싱이 유용하나, 유사하지만 약간 다른 쿼리를 실행하는 경우(예를 들어 쿼리 실행 간에 WHERE 절의 값만 변경되는 경우)에는 도움이 되지 않습니다. 이 경우 원본 테이블에서 쿼리를 실행하고 반복적으로 쿼리할 레코드를 새 대상 테이블에 작성합니다. 그런 다음 생성한 새 대상 테이블에서 쿼리를 실행합니다.

예를 들어 서로 다른 3개의 WHERE 절이 있는 쿼리 3개를 실행하려는 경우를 가정해 보겠습니다.

```
WHERE col1 = "a"
```

```
WHERE col1 = "b"
```

```
WHERE col1 = "c"
```

원본 테이블에서 쿼리를 실행하고 결과 레코드를 대상 테이블에 작성합니다.

```
SELECT col1
```

```
FROM source
```

```
WHERE col1 = "a" OR col1 = "b" OR col1 = "c"
```

'OR'로 WHERE 절을 통합하면 모든 관련 레코드를 사용할 수 있습니다. 새 대상 테이블은 원래 원본 테이블보다 훨씬 작아질 수 있습니다. BigQuery는 쿼리에서 처리되는 데이터양에 따라 비용을 청구하므로 새 대상 테이블에서 이후 쿼리를 실행하면 원본 테이블에서 직접 쿼리를 실행할 때에 비해 비용을 절감할 수 있습니다. 나중에 테이블을 정리하여 이들 테이블에 대한 저장소 비용이 누적되지 않도록 주의해야 합니다.

Tableau를 사용하여 Google BigQuery ML 모델 결과 시각화하기

BigQuery ML을 사용하면 사용자가 내장된 기계 학습 기술을 통해 BigQuery에 저장된 데이터를 기반으로 모델을 교육할 수 있습니다. 그러나 다른 데이터 작업과 마찬가지로, 모델의 결과물을 탐색하는 데 데이터베이스를 직접 쿼리하는 것이 항상 이상적인 방법은 아닙니다.

Tableau를 사용하면 BigQuery 커넥터를 통해 데이터에 대한 직관적인 이해를 돕는 방식으로 예측 모델의 결과를 쉽게 조정할 수 있습니다. 또한 Tableau를 사용하면 사용자가 모델 및 그 결과를 다른 사람과 쉽게 공유할 수 있어, 다같이 사용자 작업을 활용하고 그 이점을 누릴 수 있습니다.

하나의 사례로 [BigQuery ML과 Tableau를 이용한 주택 가격 예측](#)을 살펴보십시오.

이 사례에서는 Google BigQuery ML의 기계 학습 알고리즘을 호출하는 데 사용자 지정 SQL이 필요하지만, 다른 상황에서는 Tableau의 기본 연결 대신 사용자 지정 SQL을 사용할 때 주의해야 할 성능 관련 고려사항이 있습니다. 가능한 한, 성능 최적화를 위해 Tableau의 기본 데이터 원본 연결을 활용하십시오.

표준 SQL 쿼리를 이용한 기계 학습 모델 작성 및 실행에 대한 자세한 내용은 [Google BigQuery ML에 대한 온라인 도움말 가이드](#)를 참고하십시오.

사례 연구: Tableau와 Google BigQuery로 셀프 서비스 분석을 구축한 zulily의 유용한 팁

zulily는 비즈니스 데이터 웨어하우스로 Google BigQuery를 사용하고 데이터 액세스 및 시각적 분석에는 Tableau를 사용하여 빅 데이터 플랫폼을 구축한 빠르게 성장하고 있는 전자 상거래 회사입니다. BigQuery와 Tableau의 결합으로, 일상적인 업무에서 IT 부서의 도움 없이도, 신속하게 데이터를 수집, 처리 및 활용하여 분석이 이뤄져, 보고서 및 모델을 신속하게 작성할 수 있습니다. 또한 비즈니스 사용자가 기본적인 인사이트를 얻으려고 분석가의 도움을 받을 필요 없이, 의사 결정을 신속하게 내리는 데 사용되는 주요 데이터에 실시간으로 액세스할 수 있습니다.

다음은 zulily의 모범 사례 몇 가지입니다.

Google Compute Engine에서 Tableau Server를 사용하여 지연 시간 감소: 작업 영역이 별도의 VPC(가상 비공개 클라우드)인 기존 모델과 달리, 인터넷에 연결하지 않고 추가 설정없이 Google의 비공개 백본을 활용할 수 있습니다. 또한 과도한 프로비저닝을 할 필요 없이 배포 규모를 조정할 수 있습니다.

페더레이션된 소스 사용 및 Tableau를 BigQuery에 지정: Google Cloud 내 데이터의 경우, 외부 데이터 소스를 쿼리할 수 있고 BigQuery를 데이터 레이크로 사용하는 BigQuery 기능의 이점을 활용할 수 있습니다. 특정 상황에서는 분석을 위해 네트워크 및 Tableau로 가져와야 하는 데이터의 양을 줄이십시오.

BigQuery에서 라이브 연결로 대규모 데이터 집합 처리: 대규모 데이터 집합을 처리하고 네트워크를 통해 결과만 가져오는 BigQuery의 기능을 활용합니다. 특별한 이유가 없는 한 BigQuery에 대한 기본 Tableau 연결을 '라이브'로 설정하십시오.

자세한 내용 및 10가지 팁 목록은 2부로 구성된 다음 블로그 시리즈를 참고하십시오.

1부: zulily가 Tableau 및 Google BigQuery로 셀프 서비스 마케팅 분석 플랫폼을 만든 이유

2부: Google BigQuery와 Tableau로 셀프 서비스 분석을 구축한 zulily의 상위 10가지 팁

결론

모범 사례를 적용하면 비즈니스 사용자와 데이터 분석가 모두 Google BigQuery를 사용하여 구현한 Tableau 비주어리즘의 성능과 응답 속도를 극대화할 수 있습니다. 이러한 기술을 결합하면 사용자는 매우 신속하게 수십억 행의 데이터를 시각화할 수 있습니다.

Tableau 정보

Tableau는 완전하고, 사용하기 쉬우며, 기업에서 바로 사용할 수 있는 시각적 비즈니스 인텔리전스 플랫폼으로서, 신속하고 규모에 맞는 셀프 서비스 분석을 통해 사람들이 데이터를 보고 이해할 수 있도록 지원합니다. Tableau는 온프레미스나 클라우드, Windows나 Linux 상관없이 기존 기술 투자를 활용하고, 사용자의 데이터 환경이 변화하고 성장함에 따라 확장됩니다. 가장 가치 있는 자산인 데이터와 인적 자원을 효율적으로 활용하십시오.

추가 리소스

[Tableau 무료 평가판](#)

[Tableau 온라인 도움말 가이드: Google BigQuery](#)

[Tableau Server와 Google Cloud Platform: 클라우드에서의 신속한 비즈니스 인텔리전스](#)

[Tableau와 Google 솔루션](#)

[Tableau와 빅 데이터: 개요](#)

[클라우드에서의 비즈니스 분석 사용 이유](#)

[효율적인 통합 문서 디자인](#)

