

Daphne Kariyer online eğitim · Dijital yetkinlikler · Veri bilimi ve analitiği

Veri Bilimi ve Veri Analizi Temelleri

Modül	10
Önerilen süre	Yaklaşık 12 saat
Düzey	Başlangıç
Biçim	Online, kendi hızında
Ölçme	Daphne Kariyer ücretsiz testleri

İçindekiler

Modül 1 Veri, bilgi ve karar

Modül 2 Veri türleri ve veri kaynakları

Modül 3 Veri toplama, veri kalitesi ve temizleme

Modül 4 Temel istatistik: veriyi özetlemek ve doğru okumak

Modül 5 Elektronik tabloyla analiz: formüller ve pivot tablolar

Modül 6 SQL ile veri sorgulama

Modül 7 Veri görselleştirme ve doğru grafik seçimi

Modül 8 İş zekâsı, KPI ve gösterge panelleri

Modül 9 Veri tabanı yönetimi ve büyük veriye giriş

Modül 10 Makine öğrenmesine köprü, veri etiği ve KVKK

Kendini dene cevapları · Sözlük · Sonraki adım

Bu eğitim hakkında

Bu eğitim, işinde rapor hazırlayan ofis çalışanları ve veri analisti olmak isteyenler için hazırlandı. Önceden istatistik ya da programlama bilmen gerekmez. Elektronik tabloda basit bir toplam alabiliyorsan başlayabilirsin. Eğitim on modülden oluşur ve kendi hızında okunur. Her modülün sonunda beş soruluk bir "Kendini dene" bölümü var; cevaplar eğitimin sonunda. Kitabın sonunda bir sözlük de bulacaksın.

Modüller birbirinin üzerine kurulur. İlk dört modül veriyi anlamak, toplamak, temizlemek ve özetlemek üzerinedir. Beşinci ve altıncı modüller iki temel aracı, elektronik tabloyu ve SQL'i anlatır. Yedinci ve sekizinci modüller sonuçları grafiklerle ve gösterge panelleriyle paylaşmayı ele alır. Son iki modül veri tabanlarına, büyük veriye, makine öğrenmesine ve veri etiğine geçiş yapar. Kod örnekleri kısadır; bilgisayarında denemen önerilir ama zorunlu değildir.

Önerilen süre: yaklaşık 12 saat (okuma, örnekler ve alıştırmalar dahil). Her gün bir ya da iki modül okumak iyi bir tempodur.

Eğitimin sonunda şunları yapabileceksin:

- Bir iş sorusunu ölçülebilir bir analiz sorusuna çevirmek.
- Veri türlerini ve veri kaynaklarını ayırt etmek, bir veri setinin kalitesini değerlendirmek ve temel temizlik adımlarını uygulamak.
- Ortalama, medyan, standart sapma ve korelasyon gibi temel istatistikleri doğru yorumlamak; korelasyonu nedensellikle karıştırmamak.
- Elektronik tabloda formüller ve pivot tablolarla özet rapor hazırlamak.
- SELECT, WHERE, GROUP BY ve JOIN içeren temel SQL sorguları yazmak.
- Soruya uygun grafiği seçmek ve yanıltıcı grafiklerden kaçınmak.
- KPI tanımlamak ve sade, işe yarar bir gösterge paneli tasarlamak.
- Veri tabanı, veri ambarı ve büyük veri kavramlarını açıklamak.
- Makine öğrenmesinin analitikle ilişkisini anlatmak, kişisel verileri 6698 sayılı KVKK'ya uygun ve anonimleştirerek kullanmak.

Veri, bilgi ve karar

Her gün çok sayıda kayıt üretiyoruz: kasadan geçen her satış, depoya giren her koli, çağrı merkezine gelen her arama, web sitesindeki her tıklama. Bu kayıtların tek başına bir anlamı yoktur. Değer, kayıtlar bir soruya cevap verecek şekilde düzenlendiğinde ve bir karara yön verdiğinde ortaya çıkar. Bu modül, veri analizinin neden yapıldığını ve bir analizin hangi adımlardan geçtiğini anlatır.

Veriden karara dört basamak

Veri ile karar arasındaki ilişkiyi dört basamakla düşünmek işe yarar:

1. **Veri:** Ham kayıt. "12 Mart, Şube 4, ürün kodu 1187, adet 3, tutar 540 TL." Tek başına bir şey söylemez.
2. **Enformasyon (bilgi):** Düzenlenmiş, özetlenmiş veri. "Şube 4'ün mart ayı cirosu, şubat ayına göre düştü."
3. **İçgörü:** Enformasyonun arkasındaki neden ya da örüntü. "Düşüş, şubenin önündeki yol çalışmasının başladığı haftadan sonra başlıyor."
4. **Karar ve eylem:** İçgörüye dayanan adım. "Yol çalışması süresince şubeden ücretsiz teslimat yapalım ve sonucu iki hafta izleyelim."

Bir rapor yalnız ilk iki basamakta kalırsa okuyan kişi "peki ne yapmalıyım?" diye sorar. İyi bir analiz, en azından bir içgörü önerir ve bunun hangi kararı etkileyebileceğini söyler. Karar yine yöneticindir, ama analizin karar masasına ne getirdiği açık olmalıdır.

Veri odaklı karar verme ne demek?

Veri odaklı karar verme, kararları yalnız deneyime, alışkanlığa ya da en yüksek sesle konuşan kişinin görüşüne dayandırmak yerine, ölçülmüş kanıtla desteklemektir. Bu, sezgiyi dışlamak anlamına gelmez. Deneyimli bir satış müdürünün "bu bölgede kampanya tutmaz" sezgisi değerlidir; veri odaklı yaklaşım bu sezgiyi bir hipotez olarak ele alır ve veriyle sınar.

Veri odaklı çalışan bir ekipte şu alışkanlıklar görülür:

- Toplantıda bir iddia ortaya atıldığında "Bunu hangi veriye dayanarak söylüyoruz?" sorusu doğal karşılanır.
- Bir değişiklik yapılmadan önce mevcut durum ölçülür, böylece değişikliğin etkisi görülebilir.
- Önemli göstergelerin tanımı yazılıdır. "Aktif müşteri" herkes için aynı şeyi ifade eder.
- Veri eksik ya da şüpheliyse bu açıkça söylenir; zayıf veriyle güçlü sonuç çıkarılmaz.

Analitiğin dört türü

Analizler, cevap verdikleri soruya göre dört gruba ayrılır:

- **Betimleyici analitik (descriptive):** Ne oldu? Aylık satış raporu, şube bazında ciro, geçen yılın iade sayısı. Ofislerde hazırlanan raporların büyük kısmı bu türdür.
- **Tanısal analitik (diagnostic):** Neden oldu? İadeler hangi ürün grubunda arttı, hangi kargo firmasında gecikme yoğunlaştı? Veriyi kırılımlara ayırıp karşılaştırarak yapılır.

- **Tahmine dayalı analitik (predictive):** Ne olacak? Gelecek ayın satış tahmini, hangi müşterinin aboneliğini bırakma ihtimalinin yüksek olduğu. İstatistiksel modeller ve makine öğrenmesi kullanılır.
- **Yönlendirici analitik (prescriptive):** Ne yapmalıyız? Hangi depoda ne kadar stok tutulmalı, araçlar hangi rotayla gitmeli? Tahminleri optimizasyon ve iş kurallarıyla birleştirir.

Bu türler bir merdiven gibidir. Betimleyici analiz sağlam değilse, üzerine kurulan tahmin de sağlam olmaz. Başlangıç düzeyinde bir analistin işi çoğunlukla ilk iki basamaktadır ve bu basamaklar küçümsenecek işler değildir.

Bir analiz projesinin adımları

Analiz bir dosyayı açıp grafik çizmekle başlamaz. Tipik bir analiz şu adımlardan geçer:

1. **Soruyu tanımla:** Kim, hangi kararı vermek için neyi bilmek istiyor? Cevap ne zaman lazım?
2. **Veriyi bul ve topla:** Gerekli veri hangi sistemde? Erişim izni var mı? Hangi tarih aralığı yeterli?
3. **Veriyi incele ve temizle:** Eksik, hatalı, tekrar eden kayıtlar var mı? Alanlar doğru biçimde mi?
4. **Analiz et:** Özetle, karşılaştır, kırılımlara ayır, gerekiyorsa istatistiksel yöntem kullan.
5. **Sonucu paylaş:** Grafik, kısa rapor ya da gösterge paneli. Okuyan kişinin anlayacağı dilde.
6. **Karara bağla ve izle:** Alınan kararın etkisini ölç; gerekirse analizi güncelle.

Veri biliminde yaygın kullanılan CRISP-DM adlı süreç modeli de benzer bir döngü tarif eder: iş anlayışı, veri anlayışı, veri hazırlama, modelleme, değerlendirme ve devreye alma. İki yaklaşımda da ilk adım iş sorusunu anlamaktır ve süreç tek seferlik değil, döngüseldir.

Pratikte zamanın önemli bir kısmı üçüncü adıma, yani veriyi hazırlamaya gider. Bu bir başarısızlık işareti değildir; işin doğasıdır.

İyi bir analiz sorusu nasıl yazılır?

"Satışlar neden kötü?" bir analiz sorusu değil, bir şikâyettir. İyi bir analiz sorusu dört özellik taşır:

- **Ölçülebilir:** Neyin ölçüleceği bellidir. "Satış" ciro mu, adet mi, sipariş sayısı mı?
- **Sınırlı:** Zaman aralığı, bölge, ürün grubu bellidir.
- **Karşılaştırmalı:** Neye göre kötü? Geçen yıla, hedefe, diğer şubelere?
- **Karara bağlı:** Cevap hangi kararı etkileyecek?

Zayıf soru: "Müşteriler bizden memnun mu?"

Güçlü soru: "Son altı ayda iki ve daha fazla sipariş veren müşterilerin oranı, önceki altı aya göre değişti mi? Değiştiyse hangi şehirlerde ve hangi ürün gruplarında? Sonuç, sadakat programının bütçesini belirlemek için kullanılacak."

Veri ekosistemindeki roller

Kurumlarda veriyle çalışan farklı roller vardır. Sınırlar kurumdan kuruma değişir ama genel tablo şöyledir:

- **Veri analisti:** İş sorularını veriyle cevaplar. Elektronik tablo, SQL ve görselleştirme araçlarını kullanır. Raporlar ve gösterge panelleri hazırlar.

- **İş zekâsı (BI) uzmanı:** Kurumun raporlama altyapısını, veri modellerini ve panellerini kurar ve sürdürür.
- **Veri mühendisi:** Verinin kaynak sistemlerden alınıp temizlenmiş ve düzenli biçimde depolanmasını sağlayan veri hatlarını kurar.
- **Veri bilimci:** İstatistik ve makine öğrenmesiyle tahmin modelleri geliştirir; deney tasarlar.
- **Veri sahibi ve iş birimi:** Verinin anlamından ve kullanım kurallarından sorumlu olan, kararı veren taraf.

Raporlama yapan bir ofis çalışanı için en doğal geçiş veri analistliğidir. Bu eğitim de o yolu esas alır.

Örnek vaka

Konya'da 30 kişilik bir e-ticaret mağazası ev tekstili satıyor. Yönetici, "İadeler çok arttı, ürün kalitesi mi düştü?" diye soruyor.

Analist önce soruyu netleştiriyor: "Son üç ayda iade edilen sipariş oranı, önceki üç aya göre arttı mı? Arttıysa artış hangi ürün gruplarında, hangi iade nedenlerinde ve hangi kargo firmasında yoğunlaşıyor?"

Sonra sipariş ve iade kayıtlarını alıyor. Mutlak iade sayısının arttığını ama sipariş sayısının da arttığını görüyor. Oran olarak bakınca artış daha küçük. Kırılımlara inince artışın büyük kısmının tek bir ürün grubunda, nevresim takımlarında olduğunu, iade nedeni olarak da en çok "ölçü uymadı" seçildiğini buluyor. Ürün sayfası incelendiğinde, ay başında yeni eklenen bir ürünün ölçü bilgisinin yanlış girildiği ortaya çıkıyor.

Sonuç: Sorun genel kalite düşüşü değil, bir ürün sayfasındaki hatalı bilgi. Karar: Ürün sayfası düzeltilir, iki hafta sonra bu ürünün iade oranı yeniden ölçülür. Analiz, yönetimin aklındaki "tedarikçiyi değiştirelim" fikrinin gereksiz olduğunu da göstermiş olur.

Bu vakadaki iki ders: mutlak sayı yerine oran kullanmak ve genel tabloda kalmayıp kırılımlara inmek.

Sık yapılan hatalar

- Soruyu netleştirmeden veriye dalmak. Saatlerce grafik çizip yöneticinin aslında başka bir şey sorduğunu sonradan öğrenmek.
- Mutlak sayılarla oranları karıştırmak. Sipariş iki katına çıktıysa iade sayısının artması doğaldır.
- Yalnız ortalamaya ve toplama bakmak. Sorun çoğu zaman bir alt grupta saklıdır.
- Beklenen sonucu doğrulamak için veri aramak. Analist hipotezini çürütecek veriye de bakmalıdır.
- Raporu "ne oldu" ile bitirmek. Okuyana bulgunun hangi kararı etkileyebileceğini söylemek gerekir.
- Verinin sınırlarını gizlemek. Eksik ay, değişen tanım ya da küçük örneklem varsa rapora yazılır.

Kendini dene

1. Ham veri, enformasyon, içgörü ve karar arasındaki farkı bir örnekle açıkla.
2. "Gelecek ay hangi müşterilerin aboneliği bırakacak?" sorusu hangi analitik türüne girer?
3. "Satışlar neden kötü?" sorusunu iyi bir analiz sorusuna dönüştür.
4. Bir analiz projesinde genellikle en çok zaman hangi adıma harcanır?
5. Örnek vakada analist, iade sayısındaki artışı neden oran olarak incelemiştir?

Veri türleri ve veri kaynakları

Hangi analizi yapabileceğin, elindeki verinin türüne bağlıdır. Şehir adlarının ortalaması alınmaz, müşteri memnuniyet puanları ile sıcaklık ölçümleri aynı şekilde yorumlanmaz. Bu modülde veriyi yapısına, ölçek türüne ve kaynağına göre sınıflandırmayı öğreneceksin.

Yapılandırılmış, yarı yapılandırılmış ve yapılandırılmamış veri

- **Yapılandırılmış veri:** Satır ve sütunlardan oluşan, her sütunun türünün belli olduğu veri. Muhasebe programındaki faturalar, ERP sistemindeki stok kayıtları, bir veri tabanındaki müşteri tablosu. Analizi en kolay veri türüdür.
- **Yarı yapılandırılmış veri:** Sabit bir tablo yapısı olmayan ama etiketlerle düzenlenmiş veri. JSON ve XML dosyaları, e-faturanın XML biçimi, web servislerinin döndürdüğü yanıtlar. Her kayıta farklı alanlar bulunabilir.
- **Yapılandırılmamış veri:** Belirli bir yapısı olmayan veri. E-posta metinleri, müşteri yorumları, çağrı kayıtları, fotoğraflar, taranmış belgeler. Analiz için önce metin madenciliği, ses tanıma ya da görüntü işleme gibi yöntemlerle yapılandırılmış hale getirilmesi gerekir.

Kurumlardaki verinin önemli bir kısmı yapılandırılmamıştır, ama raporlamanın büyük kısmı yapılandırılmış veriyle yapılır.

Nitel ve nicel veri

Bir sütundaki değerler ya sayıyla ifade edilen bir miktardır ya da bir kategori ya da etikettir.

- **Nicel (sayısal) veri:** Miktar bildirir. Tutar, adet, süre, ağırlık, sıcaklık.
- **Nitel (kategorik) veri:** Bir gruba ait olmayı bildirir. Şehir, ürün grubu, ödeme yöntemi, cinsiyet, sipariş durumu.

Nicel veri kendi içinde ikiye ayrılır:

- **Kesikli:** Sayılarak elde edilir, ara değer almaz. Sipariş sayısı, çalışan sayısı, çağrı adedi. Bir günde 3,5 sipariş olmaz.
- **Sürekli:** Ölçülerek elde edilir, iki değer arasında sonsuz ara değer alabilir. Teslimat süresi, ağırlık, mesafe.

Dikkat: Bir değer rakamlardan oluşması onu sayısal yapmaz. Posta kodu, telefon numarası, TC kimlik numarası, ürün kodu rakamlardan oluşur ama bunlar etikettir. Posta kodlarının ortalamasını almak anlamsızdır. Bu alanları elektronik tabloda ya da veri tabanında metin olarak saklamak, baştaki sıfırların kaybolmasını da önler.

Dört ölçek türü

İstatistikte veriler dört ölçek türüne ayrılır. Hangi işlemin anlamlı olduğunu ölçek belirler.

1. **Sınıflama (nominal):** Kategoriler arasında sıra yoktur. Şehir, ödeme yöntemi, departman. Yalnız sayma ve oran hesaplanır. En sık görülen değer (mod) anlamlıdır.
2. **Sıralama (ordinal):** Kategoriler arasında sıra vardır ama aralıklar eşit değildir. Eğitim düzeyi, "çok memnun / memnun / kararsız / memnun değil" ölçeği, kıdem kademesi. Sıralama ve medyan

anlamlıdır. "Memnun" ile "çok memnun" arasındaki farkın, "kararsız" ile "memnun" arasındaki farka eşit olduğu söylenemez.

3. **Aralık (interval):** Aralıklar eşittir ama gerçek bir sıfır noktası yoktur. Santigrat sıcaklık, takvim yılı. 20 derece, 10 derecenin "iki katı sıcak" değildir. Fark ve ortalama anlamlıdır, oran anlamlı değildir.
4. **Oran (ratio):** Aralıklar eşittir ve gerçek sıfır vardır. Tutar, ağırlık, süre, adet. 200 TL, 100 TL'nin iki katıdır. Bütün aritmetik işlemler anlamlıdır.

İş hayatında en çok tartışma sıralama ölçeğinde çıkar. Memnuniyet anketlerinde 1–5 puanların ortalaması yaygın olarak alınır. Bu pratikte kullanılır ama yorumlarken dikkatli olunmalıdır: ortalamanın yanında her puanı veren kişilerin oranını da göstermek daha dürüst bir tablo verir.

Zaman ve konum verisi

İki veri türü özel ilgi ister:

- **Tarih ve saat:** Analizlerin büyük kısmı zamana göre yapılır. Tarihlerin gerçekten tarih türünde saklanması, saat dilimlerinin tutarlı olması ve "sipariş tarihi" ile "teslim tarihi" gibi farklı tarihlerin karıştırılmaması gerekir. Aynı tarih 03.04.2026 biçiminde bir sistemde 3 Nisan, başka bir sistemde 4 Mart olarak okunabilir.
- **Konum:** Adres, il-ilçe, posta kodu, enlem-boylam. Şehir adlarının farklı yazılması ("İstanbul", "istanbul", "IST") konum analizlerinin en sık karşılaşılan sorunudur.

Birincil ve ikincil veri

- **Birincil veri:** Belirli bir amaç için doğrudan senin ya da kurumunun topladığı veri. Müşteri anketi, saha gözlemi, deneme sonuçları. Amacına uygundur ama toplaması zaman ve emek ister.
- **İkincil veri:** Başka bir amaçla toplanmış, senin yeniden kullandığın veri. Satış sistemi kayıtları, resmî istatistikler, sektör raporları. Hızlı ve ucuzdur ama tanımları ve toplanma biçimi senin ihtiyacına tam uymayabilir.

Veri kaynakları

Bir kurumda analiz için kullanılan veri genellikle şu kaynaklardan gelir:

- **İşlem sistemleri:** ERP (kurumsal kaynak planlaması), muhasebe programı, POS (satış noktası) sistemi, depo yönetim sistemi. Günlük işlemlerin kayıtları buradadır.
- **CRM (müşteri ilişkileri yönetimi):** Müşteri kartları, görüşme notları, satış fırsatları, şikâyet kayıtları.
- **Web ve mobil analitik:** Ziyaret, tıklama, sepete ekleme, dönüşüm verileri.
- **Elektronik tablolar:** Birimlerin kendi tuttuğu listeler. Hızlıdır ama sürüm karmaşasına ve hataya açıktır.
- **Anketler ve formlar:** Çevrim içi formlar, memnuniyet anketleri.
- **Sensör ve makine verisi:** Üretim hattındaki makinelerin ölçümleri, araç takip sistemleri, akıllı sayaçlar.
- **Açık veri ve resmî istatistik:** Türkiye İstatistik Kurumu (TÜİK) yayımları, belediyelerin açık veri portalları, bakanlıkların yayımladığı veri setleri.
- **Dış veri ve API:** Döviz kurları, hava durumu, harita servisleri gibi bir uygulama programlama arayüzü (API) üzerinden alınan veriler.

Dosya biçimleri

- **CSV:** Değerlerin virgül ya da noktalı virgülle ayrıldığı düz metin dosyası. Her programda açılır. Türkçe karakterler için kodlamanın (genellikle UTF-8) doğru seçilmesi gerekir; yanlış kodlamada "ş" ve "ğ" bozuk görünür.
- **XLSX:** Excel dosyası. Birden fazla sayfa, biçimlendirme ve formül içerebilir. Birleştirilmiş hücreler ve renkle işaretlenmiş bilgiler analiz için sorun yaratır.
- **JSON:** Web servislerinin yaygın kullandığı yarı yapılandırılmış biçim.
- **Veri tabanı tabloları:** SQL ile sorgulanan, ilişkili tablolar. Altıncı modülde ayrıntılı ele alınacak.

Uygulama

Gaziantep'te 15 şubeli bir eczane zincirinin satış dosyasında şu sütunlar var: satış tarihi, şube kodu, şube ili, ürün barkodu, ürün kategorisi, adet, birim fiyat, ödeme yöntemi, müşteri yaş grubu (18–30, 31–45, 46–60, 60 üstü), müşteri memnuniyet puanı (1–5).

Her sütun için türü ve ölçeği belirle, sonra kendi cevabınla karşılaştır:

1. Satış tarihi: zaman verisi; aralık ölçeğinde düşünülür. Gün farkı hesaplanabilir.
2. Şube kodu: rakamlardan oluşsa da nitel, sınıflama ölçeği. Metin olarak saklanmalı.
3. Şube ili: nitel, sınıflama. Yazım birliği kontrol edilmeli.
4. Ürün barkodu: nitel, sınıflama. Sayı olarak saklanırsa uzun barkodlar bilimsel gösterime dönüşüp bozulabilir.
5. Ürün kategorisi: nitel, sınıflama.
6. Adet: nicel, kesikli, oran ölçeği.
7. Birim fiyat: nicel, sürekli kabul edilir, oran ölçeği.
8. Ödeme yöntemi: nitel, sınıflama.
9. Müşteri yaş grubu: nitel, sıralama. Gruplar sıralıdır ama ortalaması alınamaz.
10. Memnuniyet puanı: sıralama ölçeği. Ortalama verilecekse puan dağılımı da gösterilmeli.

Bu sınıflamayı yapmak, sonraki adımda hangi özeti ve hangi grafiği kullanacağını belirler.

Sık yapılan hatalar

- Kod ve numara alanlarını sayı olarak saklamak. Baştaki sıfırlar kaybolur, uzun numaralar yuvarlanır.
- Sıralama ölçeğindeki puanları yalnız ortalamayla raporlamak.
- Farklı sistemlerden gelen aynı alanın aynı anlama geldiğini varsaymak. Bir sistemde "müşteri" fatura kesilen firma, diğerinde teslimat yapılan şube olabilir.
- CSV dosyasını açarken kodlamayı ve ayırıcıyı kontrol etmemek. Türkçe karakterler ve ondalık virgül bu yüzden bozulur.
- İkincil verinin tanımını okumadan kullanmak. Resmî istatistiklerde yöntem notları sonucu doğrudan etkiler.
- Renkle ya da birleştirilmiş hücrelerle bilgi saklamak. Renk bir veri değildir; ayrı bir sütun olarak yazılmalıdır.

Kendini dene

1. Müşteri yorumları ve çağrı merkezi ses kayıtları hangi veri yapısına girer?
2. Posta kodunun ortalamasını almak neden anlamsızdır?
3. "Lise, ön lisans, lisans, yüksek lisans" değerleri hangi ölçek türündedir?
4. Santigrat sıcaklık neden oran ölçeği değil de aralık ölçeğidir?
5. Birincil ve ikincil verinin birer avantajını ve dezavantajını yaz.

Veri toplama, veri kalitesi ve temizleme

Analizin kalitesi, kullanılan verinin kalitesini geçemez. Hatalı bir veriyle yapılan çok iyi bir analiz, yanlış bir karara güzel bir gerekçe üretir. Bu modülde verinin nasıl daha temiz toplanabileceğini, kalitesinin nasıl ölçüleceğini ve en sık görülen sorunların nasıl temizleneceğini öğreneceksin.

Veriyi toplarken kaliteyi baştan kurmak

Temizlik pahalıdır. Bu yüzden en iyi temizlik, kirliliği kaynağında önlemektir. Bir form, kayıt ekranı ya da elektronik tablo şablonu hazırlıyorsan şu ilkeler işe yarar:

- **Serbest metin yerine seçenek:** Şehir, departman, ürün grubu gibi alanlarda açılır liste kullan. "İstanbul", "İst.", "İSTANBUL" karmaşası böyle önlenir. Excel'de bunun için Veri Doğrulama (Data Validation) özelliği vardır.
- **Biçim kuralı:** Tarih alanı yalnız tarih, tutar alanı yalnız sayı kabul etsin. Telefon numarası için tek bir biçim belirle.
- **Zorunlu alan:** Analiz için vazgeçilmez alanlar boş bırakılmasın. Ama her alanı zorunlu yapmak, kullanıcıların uydurma değer girmesine yol açar; dengeyi koru.
- **Benzersiz kimlik:** Her kayda tekrar etmeyen bir numara ver. Müşteri, sipariş, ürün gibi her varlığın kendi kimliği olsun.
- **Veri sözlüğü:** Her alanın adı, anlamı, türü, birimi ve izin verilen değerleri yazılı olsun. "Tutar KDV dahil mi?" sorusunun cevabı bir yerde yazmalıdır.
- **Amaç ve ölçülülük:** Yalnız gerçekten ihtiyaç duyduğun veriyi topla. Kişisel veri söz konusuysa bu, KVKK'nın da temel ilkelerinden biridir; onuncu modülde döneceğiz.

Veri kalitesinin boyutları

Veri kalitesi tek bir şey değildir. Genellikle şu boyutlarla değerlendirilir:

- **Doğruluk:** Kayıt gerçeği yansıtıyor mu? Müşterinin adresi gerçekten bu mu?
- **Eksiksizlik:** Gerekli alanlar dolu mu? Siparişlerin ne kadarında teslim tarihi boş?
- **Tutarlılık:** Aynı bilgi farklı yerlerde aynı mı? CRM'deki müşteri sayısı ile muhasebedeki cari sayısı neden farklı?
- **Geçerlilik:** Değer izin verilen biçimde ve aralıkta mı? Yaşı 212 olan müşteri, 31 Şubat tarihli fatura geçersizdir.
- **Benzersizlik:** Aynı kayıt birden fazla kez girilmiş mi?
- **Güncellik:** Veri karar için yeterince yeni mi? Altı ay önceki stok listesiyle bugünün siparişi planlanmaz.

Bir veri setini teslim aldığında bu altı boyut için kısa bir kontrol yapmak, sonradan çıkacak sürprizlerin çoğunu önler.

İlk bakış: veriyi tanımak

Temizliğe başlamadan önce veriyi tanı. Şu soruların cevabını not al:

1. Kaç satır ve kaç sütun var? Beklediğin sayıya yakın mı?

2. Bir satır neyi temsil ediyor: bir sipariş mi, bir sipariş kalemi mi, bir müşteri mi?
3. Her sütunun türü doğru mu? Tarihler tarih, tutarlar sayı olarak mı okunmuş?
4. Her sütunda kaç boş değer var?
5. Sayısal sütunların en küçük ve en büyük değerleri mantıklı mı?
6. Kategorik sütunlarda kaç farklı değer var? Aynı şeyin farklı yazılışları var mı?
7. Tarih aralığı nedir? Arada eksik gün, hafta ya da ay var mı?

Bu soruların ikincisi çok önemlidir. Bir satır sipariş kalemiyse ve sen satırları sayıp "sipariş sayısı" dersen, birden fazla ürün içeren siparişleri birden fazla kez saymış olursun.

En sık görülen sorunlar ve çözümleri

Tekrar eden kayıtlar. Aynı sipariş iki kez aktarılmış ya da aynı müşteri iki kez açılmış olabilir. Tamamen aynı satırları silmek kolaydır. Zor olan, "Ayşe Yılmaz, Kadıköy" ile "Ayşe Yılmaz, Kadıköy" gibi küçük farklarla tekrar eden kayıtlardır. Bunlar için önce metin standartlaştırılır, sonra benzerlik kontrol edilir. Silmeden önce hangi kaydın esas alınacağına karar verilir.

Eksik değerler. Boş bir hücreyi nasıl ele alacağın, neden boş olduğuna bağlıdır. Üç temel seçenek vardır:

- **Satırı çıkarmak:** Eksik az sayıdaysa ve rastgele dağılmışsa uygundur. Eksiklik belli bir grupta yoğunsa sonucu çarpıtır.
- **Doldurmak:** Ortalama, medyan, önceki değer ya da başka bir kaynaktan doldurmak. Doldurulan değer tahmin olduğunu kayıt altına al.
- **Ayrı kategori yapmak:** "Bilinmiyor" diye işaretlemek. Kategorik alanlarda çoğu zaman en dürüst yoldur.

Dikkat: Boş ile sıfır aynı şey değildir. Satış tutarı boş olan bir gün "satış yapılmadı" anlamına gelmeyebilir; kasa sistemi o gün çalışmamış olabilir. Boşlukları sıfırla doldurmak ortalamayı aşağı çeker.

Tutarsız yazım. Baştaki ve sondaki boşluklar, büyük-küçük harf farkları, kısaltmalar. Excel'de KIRP (TRIM) fazla boşlukları siler; BÜYÜKHARF, KÜÇÜKHARF ve YAZIMDÜZENİ harf biçimini değiştirir. Türkçede bir tuzak vardır: bazı araçlar "i" ve "ı" harflerini doğru dönüştürmez. "İstanbul" büyük harfe çevrildiğinde "İSTANBUL" yerine "ISTANBUL" çıkabilir. Dönüşümden sonra sonucu mutlaka kontrol et.

Yanlış veri türü. Tutarın metin olarak okunması toplamaların sıfır çıkmasına, tarihlerin metin olması sıralamanın bozulmasına yol açar. Türkçe sayı biçiminde ondalık ayırıcı virgöl, binlik ayırıcı noktadır (1.250,50). İngilizce ayarlı bir araç bunu yanlış okuyabilir.

Aykırı değerler. Diğerlerinden çok farklı değerler. Ortalama sepeti 400 TL olan bir mağazada 85.000 TL'lik bir sipariş hata da olabilir, gerçek bir kurumsal sipariş de. Aykırı değer otomatik silinmez; önce nedeni araştırılır. Hataysa düzeltilir, gerçekse analizde ayrıca belirtilir ya da ayrı incelenir.

Geçersiz değerler. Negatif adet, gelecekteki doğum tarihi, var olmayan il kodu. Bunlar iş kurallarıyla yakalanır: "Adet sıfırdan büyük olmalı", "teslim tarihi sipariş tarihinden önce olamaz".

Temizlik kuralları

- **Ham veriye dokunma.** Orijinal dosyanın bir kopyasında çalış. Hata yaparsan geri dönebilmelisin.

- **Adımları kaydet.** Ne yaptığını yaz: "Tamamen aynı 214 satır silindi. Şehir adları standartlaştırıldı. Tutarı boş 37 sipariş analizden çıkarıldı." Bu not, raporun güvenilirliğini artırır.
- **Tekrarlanabilir yol tercih et.** Her ay aynı temizliği elle yapıyorsan, Excel'deki Power Query, SQL ya da Python gibi adımları kaydeden bir araç kullan. Power Query ile kurduğun temizlik adımları, yeni ay verisinde yenile düğmesiyle yeniden çalışır.
- **Kaynağa geri bildirim ver.** Hata sürekli aynı formdan geliyorsa, sorunu oluşturan ekrana ya da sürece bildir.

Python ile kısa bir temizlik örneği

Python'daki pandas kütüphanesi tablo biçimli veriyi işlemek için yaygın kullanılır. Aşağıdaki kod, noktalı virgülle ayrılmış bir sipariş dosyasında temel temizlik adımlarını uygular:

```
import pandas as pd

# Dosyayı oku: binlik nokta, ondalık virgül; tarih sütunu metin olarak okunur
df = pd.read_csv("siparisler.csv", sep=";", thousands=".", decimal=".",
                 dtype={"SiparisTarihi": str}, encoding="utf-8")

# Şehir adlarındaki boşlukları sil, sonra tamamen aynı satırları çıkar
df["Sehir"] = df["Sehir"].str.strip()
df = df.drop_duplicates()

# Tarihi gün.ay.yıl biçiminden çevir; geçersiz tarihler boş (NaT) olur
df["SiparisTarihi"] = pd.to_datetime(df["SiparisTarihi"], dayfirst=True,
                                     errors="coerce")

# Her sütunda kaç boş değer kaldığını göster
print(df.isna().sum())
```

Kodun her adımı Excel'de de yapılabilir. Fark, kodun bir sonraki ay aynı dosya biçimine tek seferde uygulanabilmesidir.

Uygulama

Trabzon'da bir belediyenin park ve bahçeler müdürlüğü, vatandaş başvurularını bir elektronik tabloda tutuyor. Sütunlar: başvuru numarası, başvuru tarihi, mahalle, konu, durum, kapanış tarihi. Yönetim, başvuruların ortalama kaç günde kapandığını öğrenmek istiyor.

Adım adım ilerle:

1. Dosyanın kopyasını al, orijinali ayrı klasörde sakla.
2. Başvuru numarasında tekrar var mı kontrol et. Aynı numaralı satırları incele; aynı başvurunun iki kez girildiği görülürse birini sil ve sayısını not et.
3. Mahalle adlarının farklı yazılışlarını listele. Excel'de mahalle sütununu filtreleyip benzersiz değerlere bak. Farklı yazılışları tek yazılıшта birleştir.
4. Konu alanı serbest metinse, en sık geçen konuları belirleyip bir kategori sütunu ekle: ağaç budama, park bakımı, oyun grubu onarımı, diğer.

5. Tarihlerin gerçekten tarih türünde olduğunu kontrol et. Kapanış tarihi başvuru tarihinden önce olan satırları ayıkla ve sorumlu birime sor.
6. Durumu "açık" olan başvuruların kapanış tarihi boştur. Bu doğal bir eksikliktir; ortalama kapanma süresini yalnız kapanmış başvurular üzerinden hesapla ve açık başvuru sayısını ayrıca raporla.
7. Kapanış süresi için yeni bir sütun ekle: kapanış tarihi eksi başvuru tarihi.
8. Çok uzun süreleri incele. Yıllarca açık kalmış görünen bir başvuru, sistemde kapatılmayı unutulmuş olabilir.
9. Raporun altına temizlik notunu ekle.

Bu adımlardan sonra yalnız "ortalama 11 gün" demek yerine, medyanı, konu bazında süreleri ve açık başvuru sayısını da gösterebilirsin. Medyanın neden önemli olduğunu bir sonraki modülde göreceksin.

Sık yapılan hatalar

- Ham dosya üzerinde çalışmak ve geri dönüş yolunu kaybetmek.
- Boş hücreleri sıfırla doldurmak.
- Aykırı değerleri nedenini araştırmadan silmek.
- Bir satırın neyi temsil ettiğini anlamadan saymak. Sipariş kalemelerini sipariş sanmak.
- Temizlik adımlarını yazmamak. Aynı rapor bir ay sonra farklı sonuç verdiğinde nedenini kimse bilmez.
- Aynı temizliği her ay elle tekrar etmek. Hem zaman kaybıdır hem de her seferinde farklı hata yapma riski taşır.
- Türkçe karakter dönüşümünü kontrol etmemek.

Kendini dene

1. Veri kalitesinin altı boyutundan dördünü say.
2. Bir formda şehir alanını serbest metin yerine açılır liste yapmak hangi sorunu önler?
3. Satış tutarı boş olan günleri sıfırla doldurmak neden yanlış olabilir?
4. 85.000 TL'lik tek bir sipariş gördüğünde ilk ne yaparsın?
5. Temizlik adımlarını yazılı tutmanın iki faydasını söyle.

Temel istatistik: veriyi özetlemek ve doğru okumak

İstatistik, çok sayıda değeri birkaç anlamlı sayıyla özetlemeye ve bu özetlerden ne kadar emin olabileceğimizi söylemeye yarar. Bir analistin her gün kullandığı istatistik bilgisi düşündüğünden azdır, ama bu azı doğru kullanmak çok önemlidir. Bu modül formüllerden çok yoruma odaklanır.

Merkezi eğilim: ortalama, medyan, mod

Bir e-ticaret mağazasında bir sabah gelen yedi siparişin tutarları şöyle olsun: 150, 180, 200, 220, 250, 300 ve 4.200 TL.

- **Ortalama (aritmetik ortalama):** Değerlerin toplamı, değer sayısına bölünür. Toplam 5.500 TL, ortalama yaklaşık 786 TL.
- **Medyan (ortanca):** Değerler küçükten büyüğe sıralandığında tam ortadaki değer. Burada 220 TL. Değer sayısı çiftse ortadaki iki değer ortalaması alınır.
- **Mod (tepe değer):** En sık görülen değer. Kategorik verilerde en çok kullanılan özettir: en çok tercih edilen ödeme yöntemi gibi.

Yedi siparişin altısı 300 TL ya da altında, ama ortalama 786 TL. Tek bir büyük sipariş ortalamayı yukarı çekmiştir. Medyan ise bu uç değerden etkilenmez ve "tipik sipariş" hakkında daha doğru bir fikir verir.

Genel kural: Dağılım dengeliyse ortalama ile medyan birbirine yakındır. Gelir, maaş, sepet tutarı, ev fiyatı, çözüm süresi gibi birkaç çok büyük değer içeren verilerde medyan daha temsildir. İkisini birlikte raporlamak ve aralarındaki farkı yorumlamak iyi bir alışkanlıktır.

Yayılım: veriler ne kadar dağınık?

İki kargo deposunun ortalama sevkiyat süresi 2 gün olsun. Birinde siparişlerin neredeyse hepsi 1,5 ile 2,5 gün arasında çıkıyor. Diğerinde siparişlerin bir kısmı aynı gün, bir kısmı 5 gün sonra çıkıyor. Ortalama aynı, müşteri deneyimi çok farklı. Bu farkı yayılım ölçüleri gösterir:

- **Açıklık (range):** En büyük değer eksi en küçük değer. Hesaplaması kolaydır ama tek bir uç değerden çok etkilenir.
- **Varyans:** Değerlerin ortalamadan farklarının karelerinin ortalaması. Birimi karesel olduğu için doğrudan yorumlanması zordur.
- **Standart sapma:** Varyansın karekökü. Verinin birimiyle ifade edilir ve "değerler ortalamadan tipik olarak ne kadar uzaklaşıyor?" sorusuna cevap verir. Excel'de örneklem için STDSAPMA.S (STDEV.S) fonksiyonu kullanılır.
- **Çeyrekler:** Sıralanmış veriyi dört eşit parçaya bölen değerler. Birinci çeyrek (Ç1) verinin yüzde 25'inin altında kaldığı değer, üçüncü çeyrek (Ç3) yüzde 75'inin altında kaldığı değerdir. Medyan ikinci çeyreklerdir.
- **Çeyrekler arası açıklık (IQR):** Ç3 eksi Ç1. Verinin ortadaki yarısının ne kadar yayıldığını gösterir ve uç değerlerden etkilenmez.

Aykırı değerleri işaretlemek için yaygın kullanılan bir kural vardır: Ç1'den $1,5 \times \text{IQR}$ kadar aşağıda ya da Ç3'ten $1,5 \times \text{IQR}$ kadar yukarıda kalan değerler aykırı değer adaydır. Bu bir işaretleme kuralıdır, silme kuralı değildir.

Dağılımın şekli

Değerlerin hangi aralıkta ne sıklıkla görüldüğüne dağılım denir ve histogramla görülür.

- **Normal dağılım:** Çan eğrisi biçiminde, simetrik dağılım. Ortalama, medyan ve mod aynı noktadadır. Değerlerin yaklaşık yüzde 68'i ortalamadan bir standart sapma, yaklaşık yüzde 95'i iki standart sapma, yaklaşık yüzde 99,7'si üç standart sapma uzaklık içinde kalır. Üretimdeki ölçüm hataları ve birçok fiziksel ölçüm bu şekle yakındır.
- **Sağa çarpık dağılım:** Değerlerin çoğu düşüktür, birkaç çok yüksek değer sağa doğru uzun bir kuyruk oluşturur. Gelir, sepet tutarı, müşteri başına sipariş sayısı genellikle böyledir. Ortalama medyandan büyük çıkar.
- **Sola çarpık dağılım:** Uzun kuyruk solda, düşük değerlerdedir. Kolay bir sınavdaki puanlar örnek verilebilir.

İş verisinin önemli bir kısmı normal dağılmaz. Bu yüzden "ortalamanın iki standart sapma üstü" gibi kuralları her veride körü körüne uygulamak yanlıtabilir. Önce histograma bak.

Yüzde ve yüzde puan

Bir web sitesinin dönüşüm oranı yüzde 2'den yüzde 3'e çıktıysa:

- Artış **1 yüzde puandır** (3 eksi 2).
- Artış **yüzde 50'dir** (1 bölü 2).

İki ifade de doğrudur ama farklı şey söyler. Raporda hangisini kullandığını açıkça yaz. "Dönüşüm yüzde 50 arttı" ifadesi, küçük bir oranın küçük bir değişimini olduğundan büyük gösterebilir.

Korelasyon nedensellik değildir

Korelasyon, iki değişkenin birlikte değişme eğilimidir. Pearson korelasyon katsayısı -1 ile $+1$ arasında değer alır:

- **+1'e yakın:** Biri artarken diğeri de güçlü biçimde artar.
- **-1'e yakın:** Biri artarken diğeri güçlü biçimde azalır.
- **0'a yakın:** Aralarında doğrusal bir ilişki görülmez. Doğrusal olmayan bir ilişki yine de olabilir.

Excel'de KORELASYON (CORREL) fonksiyonuyla hesaplanır.

Korelasyon, birinin diğesine neden olduğunu göstermez. Antalya'daki bir mağazada yaz aylarında hem dondurma hem de vantilatör satışları artar. İki satış arasında güçlü bir pozitif korelasyon bulunur. Ama dondurma satmak vantilatör satışını artırmaz; ikisini de sıcak hava artırır. İki değişkeni birden etkileyen bu üçüncü etkene **karıştırıcı değişken** denir.

İki değişken arasında korelasyon gördüğünde dört olasılığı düşün:

1. A, B'ye neden oluyor.
2. B, A'ya neden oluyor (ters nedensellik). Daha çok reklam yapan şubeler daha çok satıyor olabilir; ya da çok satan şubelere daha çok reklam bütçesi veriliyor olabilir.
3. Üçüncü bir etken ikisini de etkiliyor (karıştırıcı değişken).
4. Tesadüf. Çok sayıda değişkene bakıldığında bazıları rastlantıyla birlikte hareket eder.

Nedenselliği göstermenin en güçlü yolu kontrollü deneydir. Kişiler ya da birimler rastgele iki gruba ayrılır, yalnız bir gruba değişiklik uygulanır ve sonuçlar karşılaştırılır. Web sitelerinde bu yöntem **A/B testi** denir: ziyaretçilerin yarısı eski sayfayı, yarısı yeni sayfayı görür. Rastgele atama, iki grubun diğer açılardan benzer olmasını sağlar; fark ortaya çıkarsa değişikliğe bağlanabilir.

Anakütle ve örneklem

- **Anakütle (popülasyon):** İncelemek istediğin grubun tamamı. Bir bankanın bütün bireysel müşterileri.
- **Örneklem:** Anakütleden seçilen ve gerçekten incelenen alt grup. Ankete katılan 800 müşteri.

Örneklemden anakütle hakkında sonuç çıkarılabilmesi için örneklemin anakütleyi temsil etmesi gerekir. Bunun en sağlam yolu **rastgele örneklemedir**: anakütledeki herkesin seçilme şansı bellidir. Anakütle bölge ya da yaş grubu gibi alt gruplardan oluşuyorsa her alt gruptan payı oranında seçim yapılır; buna **tabakalı örnekleme** denir.

Örneklemden hesaplanan her sonuç bir belirsizlik taşır. **Hata payı**, gerçek değer hangi aralıkta olabileceğini anlatır. Örneklem büyüdükçe hata payı küçülür, ama yavaş küçülür: hata payını yarıya indirmek için örneklemini yaklaşık dört katına çıkarmak gerekir.

En önemli nokta şudur: Örneklemi büyütmek **yanlılığı** düzeltmez. Yalnız şikâyet etmek isteyenlerin doldurduğu bir anket, bin kişiyle de on bin kişiyle de memnuniyeti olduğundan düşük gösterir. Sık görülen yanlılık türleri:

- **Seçim yanlılığı:** Örneklemin seçilme biçimi bazı grupları dışarıda bırakır. Yalnız mobil uygulama kullanıcılarına yapılan bir anket, uygulamayı kullanmayan yaşlı müşterileri temsil etmez.
- **Yanıt vermeme yanlılığı:** Ankete cevap verenler cevap vermeyenlerden farklıdır.
- **Hayatta kalma yanlılığı:** Yalnız ayakta kalan örneklere bakmak. Yalnız hâlâ çalışan müşterilerle konuşup iş bırakanların nedenlerini hiç duymamak.

İstatistiksel anlamlılık kısaca

İki grup arasında bir fark gördüğünde, bunun gerçek bir fark mı yoksa rastlantı mı olduğunu sınamak için istatistiksel testler kullanılır. Bir farkın "istatistiksel olarak anlamlı" olması, rastlantıyla açıklanmasının düşük olasılıklı olduğu anlamına gelir. Ama anlamlı fark her zaman önemli fark değildir. Çok büyük örneklemlerde çok küçük farklar da anlamlı çıkar; o farkın iş açısından değerli olup olmadığına ayrıca bakılır.

Uygulama

Eskişehir'de bir hastanenin poliklinik bekleme süreleri (dakika) bir gün için şöyle: 8, 10, 12, 12, 15, 18, 20, 22, 25, 95.

1. Ortalamayı hesapla: toplam 237, ortalama 23,7 dakika.
2. Medyanı bul: on değer var, ortadaki iki değer 15 ve 18, medyan 16,5 dakika.
3. Modu bul: 12 dakika iki kez görülüyor.
4. Açıklığı hesapla: 95 eksi 8, 87 dakika.
5. Yorumla: Hastaların çoğu 25 dakika ya da daha az bekliyor. 95 dakikalık tek bir değer ortalamayı medyanın üzerine çekiyor. Bu değer bir sistem hatası mı, gerçekten uzun bekleyen bir hasta mı, araştırılmalı.

6. Rapora "Ortalama bekleme 23,7 dakika" yerine "Medyan bekleme 16,5 dakika; bir hasta 95 dakika bekledi, nedeni inceleniyor" yazmak daha doğru bir tablo verir.

Tek günlük veriyle genel sonuç çıkarılmaz. Daha sağlam bir değerlendirme için birkaç haftalık veri, gün ve saat kırılımıyla incelenmelidir.

Sık yapılan hatalar

- Çarpık veride yalnız ortalamayı raporlamak.
- Yayılımı hiç göstermemek. Aynı ortalama çok farklı durumları gizleyebilir.
- Yüzde ile yüzde puanı karıştırmak.
- Korelasyonu nedensellik gibi sunmak. "Eğitim alan çalışanlar daha verimli, demek ki eğitim verimliliği artırıyor" sonucu, zaten verimli çalışanların eğitime seçilmiş olma ihtimalini görmezden gelir.
- Küçük ya da yanlış bir örneklemden genel sonuç çıkarmak.
- Örnekleme büyütmeyi yanlışlığı düzelttiğini sanmak.

Kendini dene

1. 150, 180, 200, 220, 250, 300 ve 4.200 değerlerinde neden medyan ortalama dan daha temsilidir?
2. Ortalamaları aynı olan iki deponun farkını hangi ölçüyle gösterirsin?
3. Dönüşüm oranı yüzde 4'ten yüzde 5'e çıktı. Artışı yüzde puan ve yüzde olarak ifade et.
4. Dondurma ve vantilatör satışlarının birlikte artmasını açıklayan etkene ne ad verilir?
5. Bir anketin örneklemini büyütmek seçim yanlışlığını düzeltir mi? Neden?

Elektronik tabloyla analiz: formüller ve pivot tablolar

Microsoft Excel ve Google E-Tablolar gibi elektronik tablolar, ofislerde en yaygın analiz aracıdır. Birkaç yüz bin satıra kadar olan verilerle hızlı özet çıkarmak, kontrol yapmak ve rapor hazırlamak için çok uygundur. Bu modülde analize uygun tablo yapısını, en çok kullanılan formülleri ve pivot tabloyu öğreneceksin. Fonksiyon adları Türkçe Excel'e göre yazıldı; İngilizce karşılıkları parantez içinde. Türkçe ayarlı Excel'de formül bağımsız değişkenleri noktalı virgülle ayrılır.

Analize uygun tablo yapısı

Bir tablonun analiz edilebilir olması için şu kurallara uyması gerekir. Bu yapıya düzenli veri (tidy data) da denir:

- Her satır tek bir kaydı temsil eder: bir sipariş, bir çalışan, bir sevkiyat.
- Her sütun tek bir değişkeni taşır: tarih, şehir, tutar.
- Tek bir başlık satırı vardır; başlıklar kısa ve benzersizdir.
- Birleştirilmiş hücre, ara toplam satırı, boş satır ve boş sütun yoktur.
- Aylar sütun başlığı olarak yan yana dizilmez. "Ocak, Şubat, Mart" sütunları yerine bir "Ay" sütunu ve bir "Tutar" sütunu kullanılır.
- Bir hücrede tek bir bilgi vardır. "Ankara / Çankaya" yerine il ve ilçe ayrı sütunlardadır.

Raporlama için hazırlanmış, renkli, ara toplamalı tablolar okumak için güzeldir ama analiz için kötüdür. Ham veriyi düzenli tutup raporu ondan üretmek doğru yoldur.

Excel'de veri aralığını seçip Ctrl+T ile **tablo olarak biçimlendirmek** iyi bir başlangıçtır. Tablo yeni satır eklendikçe kendiliğinden genişler, formüller sütun boyunca kopyalanır ve pivot tablolar yeni satırları da kapsar.

Görelili ve mutlak başvuru

Bir formülü aşağı ya da yana kopyaladığında hücre başvuruları kayar. C2 hücresindeki " $=A2+B2$ " formülü bir alt satıra kopyalandığında " $=A3+B3$ " olur. Buna **görelili başvuru** denir.

KDV oranı gibi sabit bir değeri tek bir hücrede tutuyorsan, o hücreye yapılan başvurunun kaymaması gerekir. Bunun için sütun harfinin ve satır numarasının önüne dolar işareti konur: $\$E\$1$. Buna **mutlak başvuru** denir. Aşağıdaki örnekte A sütununda KDV hariç tutarlar, E1 hücresinde KDV oranı var:

B2 hücresi: $=A2*(1+\$E\$1)$
 B3 hücresi: $=A3*(1+\$E\$1)$ (B2 aşağı kopyalanınca)

Formül kopyalandığında A2 başvurusu A3'e kayar, $\$E\$1$ sabit kalır. F4 tuşu başvuru türleri arasında geçiş yapar.

En çok kullanılan formüller

Özet fonksiyonları:

- TOPLA (SUM), ORTALAMA (AVERAGE), ORTANCA (MEDIAN), MİN (MIN), MAK (MAX).
- EĞERSAY ile koşullu sayma aşağıda anlatılıyor; düz sayma için de COUNT ve COUNTA fonksiyonlarının Türkçe karşılıkları kullanılır. COUNT yalnız sayı içeren hücreleri, COUNTA bütün dolu hücreleri sayar.

Koşullu özetler: Bir koşula uyan kayıtları saymak ya da toplamak için kullanılır.

- =EĞERSAY(C:C; "İzmir") C sütununda "İzmir" yazan hücreleri sayar.
- =ETOPLA(C:C; "İzmir"; F:F) C sütunu "İzmir" olan satırların F sütunundaki tutarlarını toplar.
- =ÇOKETOPLA(F:F; C:C; "İzmir"; D:D; "Tekstil") birden fazla koşul kullanır. Dikkat: ÇOKETOPLA'da toplanacak aralık en başa yazılır.
- ÇOKEĞERSAY (COUNTIFS) ve ÇOKEĞERORTALAMA (AVERAGEIFS) aynı mantıkla çalışır.

Koşul fonksiyonu:

- =EĞER(F2>=1000; "Büyük sipariş"; "Standart") tutara göre bir etiket üretir.
- =EĞERHATA(G2/H2; 0) bölme hatası olursa 0 yazar. Hatayı gizlemek yerine nedenini anlamak her zaman daha iyidir; EĞERHATA'yı bilinçli kullan.

Arama fonksiyonları: İki tabloyu ortak bir alan üzerinden eşleştirmek için kullanılır. Örneğin sipariş tablosuna ürün tablosundan kategori bilgisini getirmek.

- =DÜŞEYARA(A2; Urunler!A:C; 3; YANLIŞ) A2'deki ürün kodunu Urunler sayfasının A sütununda arar ve aynı satırdaki üçüncü sütunun değerini getirir. Son bağımsız değişken YANLIŞ, tam eşleşme demektir; unutulursa yanlış satırdan değer gelebilir.
- =ÇAPRAZARA(A2; Urunler!A:A; Urunler!C:C; "Bulunamadı") aynı işi yapar. Daha yeni Excel sürümlerinde bulunur; aranan sütunun solda olması gerekmez ve bulunamayan değer için ne yazılacağı belirlenebilir.

Arama sonucu #YOK (#N/A) çıkıyorsa aranan değer diğer tabloda yoktur. En sık nedenler: fazladan boşluk, sayı ile metin olarak saklanan kodların eşleşmemesi ve gerçekten eksik kayıt.

Metin ve tarih fonksiyonları:

- KIRP (TRIM) fazla boşlukları siler.
- BİRLEŞTİR (CONCATENATE) ya da & işareti metinleri birleştirir.
- SOLDAN (LEFT), SAĞDAN (RIGHT), PARÇAAL (MID) metnin bir parçasını alır.
- YIL (YEAR), AY (MONTH), GÜN (DAY) tarihten yıl, ay ve günü çıkarır. Aylık özet için yardımcı sütun olarak kullanışlıdır.
- METNEÇEVİR (TEXT) bir sayıyı ya da tarihi istenen biçimde metne çevirir.

Pivot tablo

Pivot tablo (Türkçe sürümlerde PivotTable ya da Özet Tablo), bir veri tablosunu sürük-le-bırak yöntemiyle özetleyen araçtır. Formül yazmadan "şehir ve ay bazında toplam satış" gibi bir özeti saniyeler içinde üretir. Analistin en çok kullandığı Excel özelliklerinden biridir.

Pivot tablonun dört alanı vardır:

- **Satırlar:** Özeti satırlarında görünecek kategori. Örneğin şehir.
- **Sütunlar:** Sütunlarda görünecek kategori. Örneğin ay.
- **Değerler:** Hesaplanacak sayı. Örneğin tutar toplamı, sipariş sayısı.
- **Filtreler:** Tüm özeti süzen alan. Örneğin yalnız "Teslim edildi" durumundaki siparişler.

Değerler alanında varsayılan hesap toplamdır. Alan ayarlarından sayı, ortalama, en büyük, en küçük gibi hesaplara geçebilirsiniz. "Değerleri farklı göster" seçeneğiyle her hücreyi genel toplamın ya da satır toplamının yüzdesi olarak göstermek mümkündür. Tarih alanlarını ay, çeyrek ve yıla göre gruplayabilir; **dilimleyici** (slicer) ekleyerek okuyucunun filtreleri düğmeyle değiştirmesini sağlayabilirsiniz.

Pivot tablo kaynak veri değiştiğinde kendiliğinden güncellenmez. Veriyi güncelledikten sonra **Yenile** komutunu çalıştırmak gerekir. Bu adım unutulduğunda rapor eski rakamları gösterir.

Koşullu biçimlendirme

Koşullu biçimlendirme, değerleri renk, veri çubuğu ya da simgeyle işaretler. Hedefin altında kalan şubeleri kırmızıya boyamak, en yüksek on değeri vurgulamak gibi. Okuyucunun gözünü önemli noktaya yönlendirir. Ama her hücreyi renklendirmek hiçbir şeyi vurgulamamak demektir; az ve amaçlı kullan.

Elektronik tablonun sınırları

Elektronik tablolar esnektir, ama bu esneklik hata riskini de getirir:

- Bir hücreye elle yazılmış sayı, formülün yerini sessizce alabilir.
- Aynı dosyanın e-postayla dolaşan farklı sürümleri karışır.
- Satır sayısı çok büyüdükçe dosya yavaşlar. Excel'in bir sayfadaki satır sınırı 1.048.576'dır.
- Kimin neyi ne zaman değiştirdiği izlenemeyebilir.

Bu sınırlara yaklaşıldığında veri tabanı, SQL ve iş zekâsı araçlarına geçilir. Sonraki modüller bu araçları anlatır.

Uygulama

Mersin'de 60 kişilik bir lojistik firması, sevkiyat kayıtlarını Excel'de tutuyor. Sütunlar: sevkiyat no, çıkış tarihi, varış ili, araç tipi, müşteri, ağırlık (kg), navlun tutarı, teslim durumu, gecikme günü. Operasyon müdürü, hangi illere yapılan sevkiyatlarda gecikmenin yoğunlaştığını soruyor.

1. Veri aralığını Ctrl+T ile tablo yap. Başlıkların tek satır olduğunu, birleştirilmiş hücre bulunmadığını kontrol et.
2. Gecikme günü I sütunundaysa, yeni bir yardımcı sütunun ilk satırına =EĞER(I2>0; 1; 0) yaz ve aşağı kopyala. Bu sütun geciken sevkiyatlarda 1, diğerlerinde 0 olur. Excel tablosunda formül sütun boyunca kendiliğinden kopyalanır.
3. Ekle menüsünden pivot tablo oluştur ve yeni bir sayfaya yerleştir.
4. Varış ilini Satırlar alanına sürükle.
5. Sevkiyat numarasını Değerler alanına sürükle ve hesabı "Say" yap. Bu, ile göre sevkiyat sayısını verir.

6. Yardımcı sütunu da Değerler alanına sürükley; hesap "Ortalama" olsun. 0 ve 1'lerin ortalaması, geciken sevkiyat oranını verir. Sayı biçimini yüzde yap.
7. Gecikme oranına göre büyükten küçüğe sırala.
8. Az sevkiyat yapılan illere dikkat et. Ayda 3 sevkiyatın 2'si gecikmişse oran yüksek görünür ama bu güvenilir bir sonuç değildir. Sevkiyat sayısı belirli bir eşiğin altındaki illeri ayrıca işaretle.
9. Çıkış tarihini Sütunlar alanına ekleyip aya göre grupla. Gecikmenin belirli aylarda mı yoğunlaştığını gör.
10. Araç tipi için bir dilimleyici ekle. Gecikmenin belirli bir araç tipinde mi toplandığını kontrol et.

Bu çalışma, "gecikme fazla" şikâyetini "Şu üç ilde, kış aylarında, şu araç tipinde gecikme oranı diğerlerinin belirgin şekilde üstünde" bulgusuna çevirir.

Sık yapılan hatalar

- Analiz için rapor görünümü, birleştirilmiş hücreli tabloları kullanmak.
- DÜŞEYARA'da son bağımsız değişkeni boş bırakmak ve yaklaşık eşleşmeyle yanlış değer getirmek.
- Formül sonuçlarının üzerine elle sayı yazmak.
- Sabit değerlere mutlak başvuru kullanmamak; formül kopyalandığında başvurunun kayması.
- Pivot tabloyu yenilemeyi unutmak.
- Küçük gruplarda oranları büyük gruplarla aynı güvenle yorumlamak.
- Her hata mesajını EĞERHATA ile gizlemek; gerçek veri sorunlarını görünmez yapmak.

Kendini dene

1. Aylara ait satışların "Ocak, Şubat, Mart" diye yan yana sütunlarda tutulması neden analiz için uygun değildir?
2. \$E\$1 yazımı ne anlama gelir ve ne zaman kullanılır?
3. İzmir'deki tekstil satışlarının toplamını hangi fonksiyonla bulursun?
4. Pivot tablonun dört alanını say.
5. DÜŞEYARA sonucu #YOK çıkıyorsa en sık üç neden nedir?

SQL ile veri sorgulama

Kurumların verisinin büyük kısmı ilişkisel veri tabanlarında durur. Bu veriye ulaşmanın ortak dili **SQL**'dir (Structured Query Language, yapılandırılmış sorgu dili). SQL ile milyonlarca satırlık tablolardan istediğin satırları ve özetleri birkaç satırlık bir sorguyla alırsın. Bu modülde okuma amaçlı temel sorguları öğreneceksin: SELECT, WHERE, ORDER BY, GROUP BY, HAVING ve JOIN.

Tablolar, anahtarlar ve ilişkiler

İlişkisel veri tabanında veri, birbirine bağlı tablolarda tutulur. Her tablo bir varlığı anlatır: müşteriler, siparişler, ürünler. Bu modüldeki örneklerde bir e-ticaret şirketinin iki tablosunu kullanacağız:

- **musteriler:** MusteriID, AdSoyad, Sehir, KayitTarihi
- **siparisler:** SiparisID, MusteriID, SiparisTarihi, TeslimatSehri, Tutar, Durum

Her tablonun satırlarını benzersiz biçimde tanımlayan bir sütunu vardır; buna **birincil anahtar** (primary key) denir. musteriler tablosunda MusteriID, siparisler tablosunda SiparisID birincil anahtardır.

siparisler tablosundaki MusteriID ise siparişin hangi müşteriye ait olduğunu gösterir ve musteriler tablosuna işaret eder. Başka bir tablonun birincil anahtarına işaret eden bu sütuna **yabancı anahtar** (foreign key) denir. Bir müşterinin çok sayıda siparişi olabilir; buna bire-çok ilişki denir.

Müşteri adını her siparişte tekrar yazmak yerine yalnız numarasını tutmak, verinin tek yerde güncellenmesini sağlar. Müşteri adresini değiştirdiğinde bunu tek bir satırda yaparsın.

SQL'in temel komutları bütün yaygın veri tabanı sistemlerinde (PostgreSQL, MySQL, Microsoft SQL Server, Oracle, SQLite gibi) aynıdır. Tarih fonksiyonları ve ilk N satırı alma gibi bazı ayrıntılar sistemden sisteme küçük farklar gösterir.

SELECT ve FROM: hangi sütunlar, hangi tablo

```
-- Siparişler tablosundan üç sütunu getir
SELECT SiparisID, SiparisTarihi, Tutar
FROM siparisler;
```

SELECT'ten sonra istenen sütunlar, FROM'dan sonra tablo yazılır. Bütün sütunları almak için SELECT'ten sonra yıldız işareti kullanılır. Bu, tabloyu tanımak için kullanışlıdır ama raporlarda gerekli sütunları tek tek yazmak daha doğrudur: hem daha hızlı çalışır hem de tabloya yeni sütun eklendiğinde rapor bozulmaz.

WHERE: hangi satırlar

```
-- İzmir'e 2026 yılında teslim edilen 1.000 TL ve üzeri siparişler
SELECT SiparisID, SiparisTarihi, Tutar
FROM siparisler
WHERE TeslimatSehri = 'İzmir'
    AND SiparisTarihi >= '2026-01-01'
    AND SiparisTarihi < '2027-01-01'
    AND Tutar >= 1000
    AND Durum = 'Teslim edildi'
ORDER BY Tutar DESC;
```

WHERE, koşula uyan satırları süzer. Metin ve tarih değerleri tek tırnak içinde yazılır. Sık kullanılan koşullar:

- Karşılaştırma: =, <> (eşit değil), >, <, >=, <=
- Birden fazla değer: TeslimatSehri IN ('İzmir', 'Manisa', 'Aydın')
- Aralık: Tutar BETWEEN 500 AND 1000 (iki uç dahil)
- Metin deseni: AdSoyad LIKE 'Ay%' (Ay ile başlayanlar; yüzde işareti herhangi bir karakter dizisi demektir)
- Boş değer: KapanisTarihi IS NULL. Boş değer = ile karşılaştırılmaz; "= NULL" yazmak hiçbir satır döndürmez.

ORDER BY sonucu sıralar: ASC küçükten büyüğe (varsayılan), DESC büyükten küçüğe. İlk birkaç satırı almak için PostgreSQL, MySQL ve SQLite'ta sorgunun sonuna LIMIT 10, SQL Server'da SELECT'ten hemen sonra TOP 10 yazılır.

Toplama fonksiyonları ve GROUP BY

SQL'de beş temel toplama fonksiyonu vardır: COUNT (sayı), SUM (toplam), AVG (ortalama), MIN (en küçük), MAX (en büyük). GROUP BY, satırları bir sütuna göre gruplar ve fonksiyonu her grup için ayrı hesaplar. Pivot tablonun SQL'deki karşılığı budur.

```
-- Şehir bazında teslim edilen sipariş sayısı, toplam ve ortalama tutar
SELECT TeslimatSehri,
    COUNT(*) AS SiparisSayisi,
    SUM(Tutar) AS ToplamTutar,
    AVG(Tutar) AS OrtalamaSepet
FROM siparisler
WHERE Durum = 'Teslim edildi'
GROUP BY TeslimatSehri
HAVING COUNT(*) >= 100
ORDER BY ToplamTutar DESC;
```

Bu sorguda dikkat edilecek noktalar:

- AS ile sonuç sütununa okunaklı bir ad verilir.

- SELECT'te toplama fonksiyonu dışında kalan her sütun GROUP BY'da da yer almalıdır. Burada TeslimatSehri iki yerde de var.
- **WHERE** gruplamadan önce satırları süzer. **HAVING** gruplamadan sonra grupları süzer. "100'den az siparişi olan şehirleri gösterme" koşulu bir grup koşulu olduğu için HAVING'e yazılır.
- Parantez içinde yıldız işaretiyle yazılan COUNT bütün satırları sayar. Parantez içine bir sütun adı yazılırsa, örneğin COUNT(KapanisTarihi), yalnız o sütunun boş olmayan değerleri sayılır. COUNT(DISTINCT MusteriID) ise farklı müşteri sayısını verir.

Sorgunun çalışma sırası

SQL sorgusu yazıldığı sırayla değil, mantıksal olarak şu sırayla işlenir:

1. FROM ve JOIN: tablolar alınır ve birleştirilir.
2. WHERE: satırlar süzülür.
3. GROUP BY: satırlar gruplanır.
4. HAVING: gruplar süzülür.
5. SELECT: sütunlar ve hesaplar oluşturulur.
6. ORDER BY: sonuç sıralanır.
7. LIMIT ya da TOP: ilk satırlar alınır.

Bu sıra, birçok hata mesajını açıklar. Örneğin SELECT'te verdiği bir takma adı WHERE içinde kullanamazsın, çünkü WHERE işlenirken o ad henüz oluşmamıştır.

JOIN: tabloları birleştirmek

Müşterinin adını ve kayıt şehri siparişlerle birlikte görmek için iki tabloyu ortak sütun üzerinden birleştirmen gerekir.

```
-- 2026 siparişlerini müşteri adıyla birlikte getir
SELECT m.AdSoyad, m.Sehir, s.SiparisID, s.Tutar
FROM siparisler s
INNER JOIN musteriler m
      ON s.MusteriID = m.MusteriID
WHERE s.SiparisTarihi >= '2026-01-01';
```

Burada s ve m tablolara verilen kısa takma adlardır. ON, hangi sütunların eşleşeceğini söyler.

En çok kullanılan iki birleştirme türü şunlardır:

- **INNER JOIN:** Yalnız iki tabloda da eşleşen satırları getirir. Müşteri tablosunda karşılığı olmayan bir sipariş varsa sonuca girmez.
- **LEFT JOIN:** Soldaki tablonun bütün satırlarını getirir; sağda eşleşme yoksa sağın sütunları boş (NULL) gelir.

LEFT JOIN'in pratik bir kullanımı, hiç sipariş vermemiş müşterileri bulmaktır:

```
-- Hiç sipariş vermemiş müşteriler
SELECT m.MusteriID, m.AdSoyad, m.KayitTarihi
FROM musteriler m
LEFT JOIN siparisler s
      ON s.MusteriID = m.MusteriID
WHERE s.SiparisID IS NULL;
```

Birleştirmede en sık yapılan hata, satırların beklenmedik biçimde çoğalmasıdır. Birleştirilen sütun sağ tabloda benzersiz değilse, soldaki her satır sağdaki her eşleşmeyle bir kez tekrar eder. Birleştirmeden önce ve sonra satır sayısını kontrol etmek bu hatayı yakalar.

Uygulama

Bursa'da bir yedek parça toptancısının satış müdürü şunu soruyor: "2026'nın ilk yarısında en çok ciro yaptığımız 10 müşteri kim, hangi şehirdeler ve kaç sipariş verdiler?"

Adım adım sorguyu kur:

1. Hangi tablolar gerekli? Ciro ve sipariş sayısı siparisler tablosunda, müşteri adı ve şehri musteriler tablosunda. İkisi MusteriID ile birleşir.
2. Hangi satırlar? 1 Ocak 2026 ile 30 Haziran 2026 arasındaki ve iptal edilmemiş siparişler.
3. Nasıl gruplanacak? Müşteri bazında: MusteriID, AdSoyad ve Sehir.
4. Ne hesaplanacak? Toplam ciro için SUM, sipariş sayısı için COUNT.
5. Nasıl sıralanacak? Toplam ciroya göre büyükten küçüğe; ilk 10 satır.

```
SELECT m.MusteriID, m.AdSoyad, m.Sehir,
       COUNT(*)      AS SiparisSayisi,
       SUM(s.Tutar)   AS ToplamCiro
FROM siparisler s
INNER JOIN musteriler m ON s.MusteriID = m.MusteriID
WHERE s.SiparisTarihi >= '2026-01-01'
      AND s.SiparisTarihi < '2026-07-01'
      AND s.Durum <> 'İptal'
GROUP BY m.MusteriID, m.AdSoyad, m.Sehir
ORDER BY ToplamCiro DESC
LIMIT 10;
```

Tarih aralığında bitiş için "küçüktür 1 Temmuz" yazmak, 30 Haziran'daki saatli kayıtların kaçmasını önler. SQL Server kullanıyorsan LIMIT 10 yerine SELECT TOP 10 yazarsın. Sonucu teslim etmeden önce, toplam ciroyu muhasebe raporundaki toplamla karşılaştırarak bir mantık kontrolü yap.

Sık yapılan hatalar

- Boş değerleri "= NULL" ile aramak. Doğrusu IS NULL ve IS NOT NULL.
- Toplama fonksiyonu dışındaki sütunları GROUP BY'a eklememek.
- Grup koşulunu WHERE'e, satır koşulunu HAVING'e yazmak.
- JOIN sonrası satır sayısının çoğaldığını fark etmemek ve toplamaları şişirmek.

- Tarih aralığında son günü "küçük eşittir" ile yazıp saatli kayıtları kaçırmak.
- Üretim veri tabanında, büyük tablolarda gereksiz yere bütün sütunları ve satırları çeken sorgular çalıştırmak. Okuma yetkin olsa bile sistemi yavaşlatabilirsin; mümkünse raporlama için ayrılmış kopyada çalış.

Kendini dene

1. Birincil anahtar ile yabancı anahtarın farkı nedir?
2. WHERE ile HAVING arasındaki fark nedir?
3. Kapanış tarihi boş olan kayıtları bulmak için hangi koşulu yazarsın?
4. Hiç sipariş vermemiş müşterileri bulmak için hangi JOIN türünü kullanırsın?
5. SELECT'te verdiğin takma ad neden WHERE içinde kullanılamaz?

Veri görselleştirme ve doğru grafik seçimi

İyi bir grafik, bir tabloda dakikalarca aranacak bir örüntüyü birkaç saniyede gösterir. Kötü bir grafik ise doğru veriyi yanlış anlatır. Bu modülde soruya göre grafik seçmeyi, okunaklı grafik tasarlamayı ve yanıltıcı grafiklerden kaçınmayı öğreneceksin.

Önce mesaj, sonra grafik

Grafik çizmeden önce iki soruya cevap ver:

1. **Kim okuyacak?** Genel müdür mü, operasyon ekibi mi, müşteri mi? Okuyucunun zamanı ve konuya hâkimiyeti, grafiğin ne kadar ayrıntılı olacağını belirler.
2. **Okuyucu grafikten tek bir şey hatırlayacaksa, bu ne olmalı?** "Ege bölgesindeki satışlar üç aydır düşüyor" gibi tek cümlelik bir mesaj.

Bu mesaj netse grafik türü de büyük ölçüde kendiliğinden belirlenir. Mesajı grafiğin başlığına yazmak iyi bir alışkanlıktır. "Bölgelere göre aylık satış" bir konu başlığıdır. "Ege bölgesinde satışlar üç aydır düşüyor" ise okuyucuya neye bakacağını söyleyen bir mesaj başlığıdır.

Keşif amaçlı grafikler ile sunum amaçlı grafikler de ayrılmalıdır. Analiz sırasında kendin için onlarca grafik çizersin; bunlar hızlı ve kabadır. Sunuma yalnız mesajı taşıyan birkaç grafik girer ve bunlar özenle hazırlanır.

Soruya göre grafik seçimi

- **Kategorileri karşılaştırmak:** Çubuk grafik. Şubelerin cirosu, ürün gruplarının satış adedi. Kategori adları uzunsa ya da kategori sayısı fazlaysa yatay çubuk daha okunaklıdır. Çubukları değere göre sıralamak, okuyucunun en büyük ve en küçüğü hemen görmesini sağlar.
- **Zaman içindeki değişimi göstermek:** Çizgi grafik. Aylık satış, haftalık başvuru sayısı. Zaman yatay ekseninde, soldan sağa akar. Birkaç seriyi karşılaştırmak için de uygundur; ama dört beş çizgiden sonra grafik karmaşıklaşır.
- **Parça-bütün ilişkisi:** Bütünün nasıl paylaşıldığını göstermek. Pasta grafik yalnız az sayıda parça (iki ile beş arası) varsa ve parçaların toplamı gerçekten bir bütünü oluşturuyorsa kullanılır. Göz, açıları ve alanları çubuk boylarından daha zor karşılaştırır. Daha fazla parça varsa sıralı çubuk grafik ya da yüzde 100 yığılmış çubuk daha iyi okunur.
- **Dağılımı göstermek:** Histogram. Değerleri aralıklara ayırır ve her aralıkta kaç kayıt olduğunu gösterir. Sepet tutarlarının, teslim sürelerinin dağılımı. Birden fazla grubun dağılımını karşılaştırmak için kutu grafik (box plot) kullanılır: medyanı, çeyrekleri ve aykırı değerleri tek bakışta gösterir.
- **İki sayısal değişken arasındaki ilişki:** Serpme (dağılım) grafiği. Her nokta bir kayıttır. Şubelerin metrekaresi ile cirosu, reklam harcaması ile ziyaretçi sayısı. İlişkinin yönü ve gücü görülür; aykırı noktalar dikkat çeker.
- **Coğrafi dağılım:** Harita. İllere göre renklendirilmiş haritalarda ham sayı yerine oran ya da kişi başına değer kullanılmalıdır. Aksi halde harita yalnız nüfusun nerede yoğun olduğunu gösterir.
- **Tek bir önemli sayı:** Büyük puntolu bir sayı kartı. Bu ayın cirosu, hedefe göre durum. Yanında karşılaştırma (geçen ay, hedef) olmadan tek sayı pek bir şey söylemez.

- **Ayrıntılı değerler:** Okuyucu tek tek değerlere bakacaksa, iyi düzenlenmiş bir tablo grafikten daha uygundur.

Okunaklı grafik için ilkeler

- **Çubuk grafikte eksen sıfırdan başlar.** Çubuğun boyu değeri temsil eder. Eksen 900'den başlarsa 950 ile 1.000 arasındaki küçük fark, bir çubuğun diğerinin iki katı görünmesine yol açar. Çizgi grafikte eksenin sıfırdan başlaması zorunlu değildir, çünkü çizgi değişimi gösterir; ama eksen değerleri açıkça yazılmalıdır.
- **Üç boyuttan kaçın.** 3B çubuk ve pasta grafikler perspektif nedeniyle değerleri bozar ve okumayı zorlaştırır.
- **Az renk kullan, rengi anlam için kullan.** Bütün çubukları gri yapıp yalnız mesajdaki kategoriyi vurgu rengine boyamak, okuyucunun gözünü doğru yere çeker. Aynı kategori bütün grafiklerde aynı renkte olmalıdır.
- **Renk körlüğünü düşün.** Kırmızı ile yeşili ayırt edemeyen okuyucular vardır. Yalnız kırmızı-yeşil zıtlığına dayanan grafikler bu kişiler için okunmaz. Renge ek olarak etiket ya da işaret kullan.
- **Doğrudan etiketle.** Okuyucunun gözü lejant ile grafik arasında gidip gelmesin; seri adını çizginin ucuna, değeri çubuğun üstüne yazmak çoğu zaman daha iyidir.
- **Gereksiz süsü kaldır.** Kalın ızgara çizgileri, gölge, arka plan resmi, gereksiz çerçeve. Grafikte mürekkebin çoğu veriyi göstermeye harcanmalıdır.
- **Birimi ve kaynağı yaz.** TL mi, bin TL mi, adet mi? Veri hangi tarih aralığından, hangi sistemden? Kısa bir dipnot yeterlidir.
- **Karşılaştırılacak grafiklerde ölçek aynı olsun.** Yan yana iki şube grafiğinin eksenleri farklıysa, okuyucu yanlış karşılaştırma yapar.

Yanıltıcı grafikler

Grafikler bilerek ya da bilmeyerek yanıltıcı olabilir. Şu durumlara dikkat et, hem hazırladığın hem de sana sunulan grafiklerde:

- **Kesik eksen:** Sıfırdan başlamayan çubuk eksenleri küçük farkları büyütür.
- **Seçici zaman aralığı:** Bir düşüşü gizlemek için yalnız yükselen dönemi göstermek. Grafiğin neden o tarihte başladığını sor.
- **Kümülatif grafik:** Toplam hep artar; birikimli satış grafiği, aylık satış düşse bile yükselen bir çizgi gösterir.
- **Çift eksen:** İki farklı seriyi iki ayrı dikey eksenle aynı grafiğe koymak. Eksenlerin ölçeği değiştirilerek iki seri istenildiği kadar ilişkili gösterilebilir. Mümkünse iki ayrı grafik kullan.
- **Alan ve hacim:** Değeri bir simgenin boyuyla gösterip simgeyi her yönde büyütmek. Değer iki katına çıkınca boyu ve eni iki katına çıkan simgenin alanı dört katına çıkar.
- **Toplamı yüzde 100 etmeyen pasta:** Birden fazla seçenek işaretlenebilen anket sorularında yüzdelerin toplamı 100'ü geçer; bu veri pasta grafikte gösterilmez.

Uygulama

Samsun'da 18 şubeli bir market zincirinin bölge müdürü, yönetim toplantısına bir slayt hazırlamış. Slaytta 18 dilimli, üç boyutlu bir pasta grafik var; her dilim bir şubenin yıllık cirosunu gösteriyor, renkler rastgele, lejant sağda. Toplantıda konuşulacak konu, iki yeni şubenin beklentinin altında

kalması.

Grafiği adım adım yeniden tasarla:

1. Mesajı yaz: "İki yeni şube, ilk yılında diğer şubelerin medyan cirosunun yarısına ulaşamadı."
2. Grafik türünü seç: 18 kategoriye karşılaştırmak için yatay çubuk grafik.
3. Çubukları ciroya göre büyükten küçüğe sırala.
4. Bütün çubukları açık gri yap, iki yeni şubeyi tek bir vurgu rengine boya.
5. Diğer şubelerin medyan cirosunu dikey, kesikli bir referans çizgisiyle göster ve çizginin yanına "medyan" yaz.
6. Ekseni sıfırdan başlat. Değerleri milyon TL olarak, çubukların ucuna yaz; lejantı kaldır.
7. Başlığa mesajı yaz. Altına küçük puntuyla kaynağı ve dönemi ekle.
8. Yeni şubeler yalnız birkaç aydır açıksa, yıllık ciro yerine aylık ortalama ciroyla karşılaştır. Aksi halde karşılaştırma haksız olur.

Son adım, grafik tasarımının veri mantığından ayrı düşünülemeyeceğini gösterir. En güzel grafik bile yanlış ölçüyle çizilirse yanıltır.

Sık yapılan hatalar

- Mesajı belirlemeden grafik çizmek ve programın önerdiği ilk grafik türünü kullanmak.
- Çok parçalı pasta grafikler ve üç boyutlu grafikler.
- Çubuk grafiklerde ekseni sıfırdan başlatmamak.
- Her seriye farklı, anlamsız renkler vermek; aynı kategoriye farklı grafiklerde farklı renklerle göstermek.
- Birim, dönem ve kaynak bilgisini yazmamak.
- Haritalarda oran yerine ham sayı göstermek.
- Bir slayta çok sayıda grafik sıkıştırmak. Bir slayt, bir mesaj.

Kendini dene

1. On iki aylık satış eğilimini göstermek için hangi grafik türü uygundur?
2. Pasta grafik hangi koşullarda kullanılabilir?
3. Çubuk grafikte eksenin sıfırdan başlaması neden önemlidir?
4. Sepet tutarlarının nasıl dağıldığını görmek için hangi grafiği kullanırsın?
5. "Bölgelere göre satış" başlığını mesaj taşıyan bir başlığa dönüştür.

İş zekâsı, KPI ve gösterge panelleri

İş zekâsı (business intelligence, BI), kurumun verisini toplayıp düzenleyen ve karar vericilere raporlar ve gösterge panelleri aracılığıyla sunan araçların ve süreçlerin genel adıdır. Her ay elle hazırlanan Excel raporlarının yerini, verisi kendiliğinden yenilenen ve kullanıcının kendisinin filtreleyebildiği paneller alır. Bu modülde KPI tanımlamayı, panel tasarlamayı ve en yaygın iki iş zekâsı aracını kavramsal olarak öğreneceksin.

Elle raporlamadan iş zekâsına

Birçok kurumda raporlama şöyle işler: Analist her ay başında birkaç sistemden dosya indirir, Excel'de birleştirir, temizler, pivot tablolar ve grafikler hazırlar, sonucu e-postayla gönderir. Bu yöntemin sorunları bellidir: zaman alır, her ay hata riski taşır, farklı birimler farklı rakamlar sunar ve yönetici bir ayrıntıyı merak ettiğinde yeni bir rapor istemek zorunda kalır.

İş zekâsı yaklaşımında:

- Veri kaynaklarına bir kez bağlanır; temizlik ve dönüştürme adımları kaydedilir.
- Veri belirli aralıklarla kendiliğinden yenilenir.
- Göstergelerin hesaplama yöntemi tek bir yerde tanımlanır. Herkes aynı "ciro" tanımını kullanır. Buna **tek doğru kaynak** (single source of truth) ilkesi denir.
- Kullanıcılar panel üzerinde tarih, bölge, ürün gibi filtreleri kendileri değiştirir ve ayrıntıya iner.

Analistin rolü de değişir: Aynı raporu her ay yeniden üretmek yerine, panelin doğruluğunu korur ve yeni sorulara cevap arar.

Veri modeli: olgu ve boyut tabloları

İş zekâsı araçlarında veri genellikle **yıldız şeması** (star schema) adı verilen bir yapıda modellenir:

- **Olgu tablosu (fact table):** Ölçülen olayları tutar. Her satış satırı, her sevkiyat, her çağrı. Tutar, adet, süre gibi sayısal alanlar ve boyut tablolarına bağlanan anahtarlar içerir. Genellikle çok satırlıdır.
- **Boyut tabloları (dimension tables):** Olayları tanımlayan bilgileri tutar. Tarih, müşteri, ürün, şube, çalışan. Filtreleme ve gruplama bu tablolardaki alanlarla yapılır.

Ortada olgu tablosu, çevresinde ona bağlı boyut tabloları olduğu için bu yapı yukarıdan bakıldığında bir yıldıza benzer. Ayrı bir **tarih tablosu** kullanmak özellikle önemlidir: ay, çeyrek, yıl, iş günü, resmî tatil gibi bilgiler burada tutulur ve "geçen yılın aynı dönemi" gibi karşılaştırmalar kolaylaşır.

Power BI ve Tableau

İki araç da iş zekâsı alanında yaygın kullanılır. Temel mantıkları benzerdir: veriye bağlan, modelle, görselleştir, paylaş.

Microsoft Power BI:

- **Power BI Desktop:** Raporların hazırlandığı masaüstü uygulaması.
- **Power Query:** Veriyi alma, temizleme ve dönüştürme aracı. Excel'deki Power Query ile aynı mantıkta çalışır; adımlar kaydedilir ve her yenilemede tekrar uygulanır.

- **Veri modeli ve ilişkiler:** Tablolar arasındaki bağlantılar model görünümünde kurulur.
- **DAX:** Ölçü ve hesaplanmış sütun yazmak için kullanılan formül dili.
- **Power BI hizmeti:** Raporların web üzerinde yayımlandığı, paylaşıldığı ve zamanlanmış yenilenmenin ayarlandığı çevrim içi ortam. Satır düzeyi güvenlik ile her kullanıcının yalnız yetkili olduğu veriyi (örneğin yalnız kendi bölgesini) görmesi sağlanabilir.

DAX'ta bir **ölçü**, filtre bağlamına göre yeniden hesaplanan bir formüldür. Kullanıcı panelde İzmir'i seçtiğinde "Toplam Ciro" ölçüsü yalnız İzmir için hesaplanır:

```
Toplam Ciro = SUM ( Satislar[Tutar] )
Siparis Sayisi = DISTINCTCOUNT ( Satislar[SiparisID] )
Ortalama Sepet = DIVIDE ( [Toplam Ciro], [Siparis Sayisi] )
```

DIVIDE fonksiyonu, payda sıfır olduğunda hata yerine boş değer döndürür.

Tableau:

- **Tableau Desktop:** Görselleştirmelerin hazırlandığı uygulama. Veri kaynağına bağlandıktan sonra alanlar **boyutlar** (kategorik alanlar) ve **ölçüler** (sayısal, toplanabilir alanlar) olarak ayrılır.
- **Çalışma sayfası (worksheet):** Tek bir görselleştirme. Alanlar satır ve sütun raflarına sürüklenerek grafik oluşturulur.
- **Gösterge paneli (dashboard):** Birden fazla çalışma sayfasının bir araya getirildiği ve birbirini filtreleyebildiği ekran.
- **Hikâye (story):** Panellerin ve grafiklerin bir anlatı sırasıyla dizildiği sunum.
- **Paylaşım:** Tableau Server ya da Tableau Cloud üzerinden kurum içinde; Tableau Public ile herkese açık olarak.

Hangi aracın seçileceği kurumun mevcut altyapısına, lisans politikasına ve ekibin alışkanlıklarına bağlıdır. Bir aracı iyi öğrenen kişi diğerine görece kolay geçer, çünkü veri modeli, ölçü ve panel tasarımı gibi kavramlar ortaktır.

Metrik ve KPI

Her ölçüm bir **metriktir**: ziyaretçi sayısı, çağrı sayısı, sevkiyat adedi. **KPI** (key performance indicator, temel performans göstergesi) ise bir kurumun ya da birimin hedefine ne kadar yaklaştığını gösteren, sayıca az ve özenle seçilmiş metriklerdir. Bütün KPI'lar metriktir; ama her metrik KPI değildir.

İyi bir KPI şu özellikleri taşır:

- **Bir hedefe bağlıdır.** "Müşteri sadakatini artırmak" hedefi için "tekrar sipariş veren müşteri oranı".
- **Açık tanımlıdır.** Formülü, veri kaynağı, dahil edilen ve hariç tutulan kayıtları yazılıdır.
- **Hedefi ve eşikleri vardır.** Ne zaman iyi, ne zaman dikkat, ne zaman alarm?
- **Eylem doğurur.** Gösterge kötüleştğinde kimin ne yapacağı bellidir.
- **Zamanında güncellenir.** Haftalık karar için aylık güncellenen gösterge işe yaramaz.

KPI'lar iki gruba ayrılır:

- **Gecikmeli (sonuç) göstergeler:** Olan bitenin sonucunu ölçer. Aylık ciro, yıllık müşteri kaybı oranı. Önemlidir ama değiştirmek için geç kalınmış olabilir.

- **Öncü göstergeler:** Sonucu önceden haber veren göstergeler. Satış ekibinin haftalık müşteri ziyareti sayısı, web sitesinde teklif formunu dolduranların sayısı. Erken müdahale imkânı verir.

Bir de **gösteriş metrikleri** (vanity metrics) vardır: büyük ve güzel görünen ama karara katkı sağlamayan sayılar. Toplam sosyal medya takipçisi gibi. Bunları KPI olarak izlemek, ekibin enerjisini yanlış yere yönlendirir.

KPI tanım kartı

Her KPI için kısa bir tanım kartı yazmak, aynı göstergenin farklı birimlerde farklı hesaplanmasını önler. Kartta şunlar bulunur:

1. Ad: Zamanında teslim oranı.
2. Amaç: Müşteriye söz verilen teslim tarihine uyumu izlemek.
3. Formül: Söz verilen tarihte ya da önce teslim edilen sevkiyat sayısı bölü teslim edilen toplam sevkiyat sayısı.
4. Kapsam: İptal edilen ve müşteri isteğiyle ertelenen sevkiyatlar hariç.
5. Veri kaynağı: Depo yönetim sistemindeki sevkiyat tablosu.
6. Güncelleme: Günlük.
7. Hedef ve eşikler: Hedef, yönetimin belirlediği oran; hedefin belirli bir puan altı "dikkat", daha altı "alarm".
8. Sorumlu: Operasyon müdürü.

Gösterge paneli tasarımı

Paneller amaçlarına göre üçe ayrılır:

- **Operasyonel paneller:** Günlük işi izler; sık yenilenir. Çağrı merkezindeki bekleyen çağrı sayısı, depodaki bugünkü sevkiyatlar.
- **Stratejik paneller:** Üst yönetim için; az sayıda KPI, uzun dönem eğilimi, hedefe göre durum.
- **Analitik paneller:** Analistler ve uzman kullanıcılar için; çok filtre, ayrıntıya inme imkânı.

İyi bir panelin tasarım ilkeleri:

- **Okuyucu ve soru:** Panel kimin hangi sorusuna cevap veriyor? Herkese her şeyi gösteren panel kimsenin işine yaramaz.
- **Yerleşim:** Okuyucu ekranı genellikle sol üstten başlayarak tarar. En önemli KPI kartlarını üst sıraya, eğilim grafiklerini ortaya, ayrıntı tablolarını alta yerleştir.
- **Bağlam:** Her KPI'nın yanında hedef, geçen dönem ya da geçen yılın aynı dönemi olsun. "Ciro 4,2 milyon TL" yerine "Ciro 4,2 milyon TL, hedefin yüzde 96'sı, geçen yılın aynı ayına göre yukarıda" daha anlamlıdır.
- **Sadelik:** Bir ekranda az sayıda görsel. Kaydırma gerektiren uzun paneller yerine ayrı sayfalar.
- **Tutarlı renk:** Kırmızı yalnız sorun için kullanılır. Aynı kategori her grafikte aynı renktedir.
- **Güncellik bilgisi:** "Son güncelleme" tarih ve saati panelde görünür olsun.
- **Mobil görünüm:** Yöneticiler paneli telefondan açıyorsa ayrı bir mobil yerleşim hazırla.

Örnek vaka

İzmir'de 40 kişilik bir lojistik firması, her pazartesi yönetime 12 sayfalık bir Excel raporu gönderiyor. Rapor hazırlamak analistin yaklaşık bir iş gününü alıyor ve yönetim raporun çoğunu okumuyor.

Analist şu adımlarla bir panel hazırlıyor:

1. Genel müdür ve operasyon müdürüyle kısa bir görüşme yapıp en çok hangi sorulara baktıklarını öğreniyor: Zamanında teslim ediyor muyuz? Araçlarımız dolu mu gidiyor? Hangi müşteriler kârlı?
2. Bu sorulara karşılık gelen dört KPI seçiyor ve her biri için tanım kartı yazıyor: zamanında teslim oranı, ortalama araç doluluk oranı, sevkiyat başına maliyet ve hasar bildirimli sevkiyat oranı.
3. Depo yönetim sistemi, araç takip sistemi ve muhasebe programından gelen veriyi Power Query ile alıp temizliyor; sevkiyat olgu tablosu ile müşteri, araç, güzergâh ve tarih boyut tablolarından oluşan bir yıldız şeması kuruyor.
4. Panelin üst sırasına dört KPI kartını, hedef ve geçen ay karşılaştırmasıyla yerleştiriyor. Ortaya son 12 haftanın zamanında teslim oranını gösteren bir çizgi grafik, altına güzergâh bazında ayrıntı tablosu koyuyor.
5. Tarih aralığı, müşteri ve güzergâh için filtre ekliyor; her sabah otomatik yenileme ayarlıyor.
6. İlk iki hafta panelin rakamlarını eski Excel raporuyla karşılaştırıp farkları çözüyor. Farkların bir kısmının, eski raporda iptal edilen sevkiyatların da hesaba katılmasından kaynaklandığını görüyor ve tanım kartına bunu açıkça yazıyor.

Sonuçta haftalık rapor hazırlığı büyük ölçüde ortadan kalkıyor; analist zamanını gecikmelerin nedenlerini incelemeye ayırabiliyor.

Sık yapılan hatalar

- Bir panele mümkün olan her grafiği koymak.
- KPI'ları tanımsız bırakmak; aynı adla farklı hesaplanan göstergeler.
- Hedefsiz ve karşılaştırmaz tek sayılar göstermek.
- Gösteriş metriklerini KPI gibi izlemek.
- Panelin rakamlarını kaynak sistemlerle karşılaştırmadan yayına almak.
- Yenileme hatalarını fark etmemek. Son güncelleme tarihi görünmeyen panel, eski veriyi güncel gibi gösterebilir.
- Kullanıcıya sormadan panel tasarlamak.

Kendini dene

1. Tek doğru kaynak ilkesi neyi önler?
2. Yıldız şemasında olgu tablosu ile boyut tablosunun farkı nedir?
3. Metrik ile KPI arasındaki fark nedir?
4. Öncü ve gecikmeli göstergeye birer örnek ver.
5. Bir KPI tanım kartında hangi bilgiler bulunmalıdır? En az beşini say.

Veri tabanı yönetimi ve büyük veriye giriş

Altıncı modülde veri tabanından veri okumayı öğrendin. Bu modülde perdenin arkasına bakacağız: veri tabanları veriyi nasıl güvenli ve tutarlı tutar, raporlama için neden ayrı bir veri ambarı kurulur, büyük veri ne demektir ve hangi araçlarla işlenir? Amaç bir veri tabanı yöneticisi olmak değil, bu ekiplerle aynı dili konuşabilmektir.

Veri tabanı yönetim sistemi ne yapar?

Veri tabanı yönetim sistemi (VTYS, İngilizcesi DBMS), verinin saklanması, sorgulanmasını ve korunmasını sağlayan yazılımdır. PostgreSQL, MySQL, Microsoft SQL Server ve Oracle Database yaygın ilişkisel VTYS örnekleridir. Bir elektronik tablodan farkı, şu işleri sistemli biçimde üstlenmesidir:

- **Eşzamanlı erişim:** Yüzlerce kullanıcı aynı anda okuyup yazabilir, kayıtlar birbirini bozamaz.
- **Bütünlük kuralları:** Hatalı veri girişini sistem düzeyinde engeller.
- **Yetkilendirme:** Kimin hangi tabloyu okuyabileceği ya da değiştirebileceği tanımlanır.
- **Yedekleme ve geri yükleme:** Veri belirli aralıklarla yedeklenir; bir arıza ya da hatada belirli bir ana geri dönülebilir.
- **Performans:** Büyük tablolarda bile sorguların hızlı çalışması için dizinler ve sorgu iyileştirme kullanılır.

Tablo tasarımı ve bütünlük kuralları

Tablolar oluşturulurken sütunların türü ve kuralları tanımlanır. Bu kurallar, Modül 3'te anlattığımız veri kalitesi sorunlarının bir kısmını kaynağında önler.

```
CREATE TABLE musteriler (
    MusteriID INT PRIMARY KEY,
    AdSoyad VARCHAR(100) NOT NULL,
    Sehir VARCHAR(50),
    KayitTarihi DATE NOT NULL
);

CREATE TABLE siparisler (
    SiparisID INT PRIMARY KEY,
    MusteriID INT NOT NULL,
    SiparisTarihi DATE NOT NULL,
    Tutar DECIMAL(12,2) CHECK (Tutar >= 0),
    FOREIGN KEY (MusteriID) REFERENCES musteriler (MusteriID)
);
```

Bu tanımda:

- PRIMARY KEY her satırın benzersiz olmasını sağlar; aynı sipariş numarası iki kez girilemez.
- NOT NULL alanın boş bırakılmasını engeller.

- CHECK, değerin bir koşula uymasını zorunlu kılar; negatif tutar girilemez.
- FOREIGN KEY, siparişin var olan bir müşteriye bağlanmasını zorunlu kılar. Müşteri tablosunda olmayan bir numarayla sipariş girilemez. Buna **bilgi bütünlüğü** (referential integrity) denir.
- DECIMAL(12,2), toplam 12 basamaklı ve 2'si ondalık olan bir sayı demektir. Para tutarları için kayan noktalı sayı türleri yerine bu tür tercih edilir, çünkü kuruş hesaplarında yuvarlama hatası yapmaz.

Normalizasyon, aynı bilginin birden fazla yerde tekrarlanmaması için veriyi ilişkili tablolara bölme işlemidir. Müşteri adresinin her siparişte tekrar yazıldığı bir tabloda adres değiştiğinde eski siparişlerde eski, yenilerinde yeni adres kalır ve hangisinin doğru olduğu karışır. Normalizasyon bu güncelleme sorunlarını önler. Raporlama için kurulan veri ambarlarında ise sorguların hızlı ve kolay olması için bilinçli olarak bir miktar tekrar kabul edilir; Modül 8'deki yıldız şeması buna örnektir.

SQL komut grupları

SQL komutları amaçlarına göre gruplanır:

- **Veri sorgulama:** SELECT. Analistin en çok kullandığı komut.
- **Veri işleme (DML):** INSERT (yeni satır ekleme), UPDATE (satır güncelleme), DELETE (satır silme).
- **Veri tanımlama (DDL):** CREATE (tablo oluşturma), ALTER (yapıyı değiştirme), DROP (tabloyu tamamen silme).
- **Yetki yönetimi (DCL):** GRANT (yetki verme), REVOKE (yetki geri alma).

Analistlere genellikle yalnız okuma yetkisi verilir. Bu bir güvensizlik işareti değil, **en az yetki ilkesinin** gereğidir: herkes yalnız işi için gereken erişime sahip olur. WHERE koşulu unutulmuş tek bir UPDATE ya da DELETE komutu tablonun bütün satırlarını değiştirebilir.

İşlemler ve ACID

Bir bankada bir hesaptan diğerine havale yapıldığını düşün. Para bir hesaptan düşülmeli ve diğerine eklenmelidir. İlk adımdan sonra sistem çökerse para kaybolmamalıdır. Veri tabanları bu tür işlemleri **işlem** (transaction) olarak ele alır ve dört özelliği garanti eder. Baş harfleriyle ACID denir:

- **Bölünmezlik (Atomicity):** İşlemin bütün adımları ya birlikte gerçekleşir ya da hiçbiri gerçekleşmez.
- **Tutarlılık (Consistency):** İşlem, veri tabanını kurallara uygun bir durumdan yine kurallara uygun bir duruma taşır.
- **Yalıtım (Isolation):** Aynı anda çalışan işlemler birbirinin yarım kalmış sonuçlarını görmez.
- **Kalıcılık (Durability):** Tamamlanan işlem, sistem hemen sonra çökse bile kaybolmaz.

Dizinler

Dizin (index), bir kitabın sonundaki dizin gibi çalışır. Kitabı baştan sona okumak yerine aradığın kelimenin sayfasına doğrudan gidersin. Sık sorgulanan sütunlara dizin eklemek okuma sorgularını çok hızlandırır. Bedeli, ek depolama alanı ve yazma işlemlerinin biraz yavaşlamasıdır. Bu yüzden her sütuna dizin eklenmez; hangi sorguların sık çalıştığına bakılarak karar verilir.

İşlem sistemleri ve analiz sistemleri

Kurumlarda iki farklı türde veri sistemi bulunur:

- **OLTP (çevrim içi işlem işleme):** Günlük işlemleri kaydeden sistemler. Sipariş alma, fatura kesme, stok düşme. Çok sayıda küçük ve hızlı okuma-yazma işlemi yapar. Veri normalize edilmiştir.
- **OLAP (çevrim içi analitik işleme):** Analiz ve raporlama için tasarlanmış sistemler. Az sayıda ama çok satırı tarayan büyük sorgular çalıştırır. "Son üç yılın bölge ve ürün bazında aylık cirosu" gibi.

Ağır rapor sorgularını doğrudan OLTP sistemine göndermek, müşterinin sipariş verdiği sistemi yavaşlatabilir. Bu yüzden analiz için ayrı bir yapı kurulur:

- **Veri ambarı (data warehouse):** Farklı kaynak sistemlerden gelen verinin temizlenip birleştirildiği, geçmişin saklandığı ve analiz için düzenlendiği merkezî depo. Kurumun tek doğru kaynağı olmayı hedefler.
- **ETL:** Veriyi kaynaktan çekme (Extract), dönüştürme (Transform) ve hedefe yükleme (Load) süreci. Bulut veri ambarlarının yaygınlaşmasıyla veriyi önce yükleyip sonra ambar içinde dönüştüren **ELT** yaklaşımı da sık kullanılır.
- **Veri gölü (data lake):** Yapılandırılmış, yarı yapılandırılmış ve yapılandırılmamış verinin ham biçimde, düşük maliyetle saklandığı depo. Esnektir ama düzen ve kataloglama olmazsa içinden aranan veri bulunamaz hale gelir.
- **Veri pazarı (data mart):** Veri ambarının belirli bir birime (satış, finans, insan kaynakları) yönelik alt kümesi.

NoSQL veri tabanları

Her veri tablo yapısına uymaz. NoSQL adı verilen veri tabanları farklı veri modelleri kullanır:

- **Doküman tabanlı:** Veriyi JSON benzeri belgeler olarak saklar. Ürün kataloğu gibi her kaydın farklı alanlara sahip olabileceği durumlar için uygundur. MongoDB bilinen bir örnektir.
- **Anahtar-değer:** Bir anahtara karşılık bir değer saklar; çok hızlıdır. Oturum bilgisi ve önbellek için kullanılır. Redis bilinen bir örnektir.
- **Geniş sütunlu:** Çok büyük miktarda yazma işlemini dağıtık biçimde karşılamak için tasarlanmıştır. Apache Cassandra bilinen bir örnektir.
- **Çizge (graph):** Varlıkları ve aralarındaki ilişkileri düğümler ve bağlantılar olarak saklar. Sosyal ağlar, dolandırıcılık ağlarının tespiti, öneri sistemleri. Neo4j bilinen bir örnektir.

NoSQL, SQL'in yerine geçen bir şey değildir; farklı ihtiyaçlar için farklı araçlardır. Birçok kurum ikisini birlikte kullanır.

Büyük veri

Büyük veri, geleneksel tek sunuculu veri tabanlarının ve araçların işleyemeyeceği kadar büyük, hızlı ya da çeşitli veridir. Genellikle "V" ile başlayan özelliklerle tanımlanır:

- **Hacim (Volume):** Veri miktarı tek bir makineye sığmayacak kadar büyüktür.
- **Hız (Velocity):** Veri sürekli ve hızla üretilir: sensör ölçümleri, web tıklamaları, kart işlemleri.
- **Çeşitlilik (Variety):** Tablo, metin, görüntü, ses, kayıt dosyaları bir aradadır.
- **Doğruluk (Veracity):** Verinin güvenilirliği ve kalitesi değişkendir.
- **Değer (Value):** Verinin bir işe yaraması gerekir; saklamak tek başına değer üretmez.

Büyük veri, işi birçok makineye bölerek işlenir. Bu alanda sık duyacağın araçlar:

- **Apache Hadoop:** Veriyi birçok makineye dağıtarak saklayan dosya sistemi (HDFS) ve dağıtık işleme modeliyle büyük veri alanının ilk yaygın açık kaynak çözümlerinden biri.
- **Apache Spark:** Veriyi mümkün olduğunca bellekte işleyen dağıtık işleme motoru. SQL, Python ve diğer dillerle kullanılabilir.
- **Apache Kafka:** Sürekli akan veriyi (olay akışlarını) sistemler arasında taşıyan platform.
- **Bulut veri ambarları:** Google BigQuery, Amazon Redshift, Azure Synapse Analytics ve Snowflake gibi hizmetler, altyapıyı yönetme yükünü azaltır ve SQL ile çok büyük tabloları sorgulamayı sağlar.

Analist açısından iyi haber şu: Bu araçların çoğu SQL ya da SQL'e çok benzeyen bir dille sorgulanabilir. Altıncı modülde öğrendiklerin burada da geçerlidir. Bir başka gerçek de şu: Birçok kurumun verisi "büyük veri" tanımına girmez. İyi tasarlanmış bir ilişkisel veri tabanı ya da bulut veri ambarı çoğu ihtiyacı karşılar. Büyük veri araçları, gerçek bir ihtiyaç olduğunda seçilmelidir.

Yedekleme ve güvenlik

- **Yedekleme:** Yaygın kabul gören 3-2-1 kuralı şunu önerir: verinin en az üç kopyası, en az iki farklı ortamda ve bir kopyası farklı bir konumda tutulur. Yedeğin gerçekten geri yüklenebildiği düzenli olarak denenmelidir; hiç denenmemiş yedek güvence değildir.
- **Erişim yönetimi:** Kullanıcı hesapları kişiye özel olur, ortak şifre kullanılmaz; işten ayrılan kişinin erişimi hemen kapatılır.
- **Şifreleme ve kayıt:** Hassas veri saklanırken ve iletilirken şifrelenir; kimin hangi veriye eriştiği kayıt altına alınır.
- **Standartlar:** Bilgi güvenliği yönetimi için uluslararası standart ISO/IEC 27001'dir. Kişisel veri işleniyorsa KVKK'nın veri güvenliği yükümlülükleri de geçerlidir.

Örnek vaka

Ankara'da orta ölçekli bir e-ticaret şirketinde, analistler raporlarını doğrudan sipariş veri tabanına SQL sorguları göndererek hazırlıyor. Kampanya dönemlerinde site yavaşlıyor; yazılım ekibi, ay sonunda çalışan büyük rapor sorgularının sipariş sistemini zorladığını tespit ediyor. Ayrıca pazarlama, finans ve depo ekiplerinin aynı ay için farklı ciro rakamları sunduğu görülüyor.

Şirketin attığı adımlar:

1. Raporlama için ayrı bir bulut veri ambarı kuruluyor. Sipariş, ödeme, iade ve depo verileri her gece bir ETL süreciyle ambara aktarılıyor.
2. Ciro, net satış, iade oranı gibi temel göstergelerin tanımları finans ekibiyle birlikte yazılıyor ve ambarda tek bir yerde hesaplanıyor.
3. Analistlere sipariş veri tabanında değil, ambarda okuma yetkisi veriliyor.
4. Gösterge panelleri ambara bağlanıyor; her sabah yenileniyor.
5. Gerçek zamanlı izlenmesi gereken birkaç gösterge (anlık sipariş sayısı gibi) için ayrı, hafif bir akış kuruluyor.

Sonuç olarak rapor sorguları satış sistemini etkilemez hale geliyor ve toplantılarda "hangi rakam doğru" tartışması yerini "rakam neden böyle" sorusuna bırakıyor.

Sık yapılan hatalar

- Raporlama sorgularını doğrudan canlı işlem sistemine göndermek.
- Analistlere ihtiyaçlarından fazla yetki vermek.
- WHERE koşulu olmadan UPDATE ya da DELETE çalıştırmak.
- Para tutarlarını kayan noktalı sayı türünde saklamak.
- Yedeklerin geri yüklenebildiğini hiç denememek.
- Gerçek bir ihtiyaç yokken büyük veri araçlarına geçmek; ya da veri gölünü kataloglamadan büyütüp kullanılamaz hale getirmek.
- Dizinleri ölçmeden eklemek ya da hiç eklememek.

Kendini dene

1. VTYS'nin elektronik tabloya göre üstlendiği üç işi say.
2. FOREIGN KEY kısıtı hangi hatayı önler?
3. ACID'in dört harfi neyi ifade eder?
4. OLTP ve OLAP sistemlerinin farkı nedir?
5. Veri ambarı ile veri gölü arasındaki temel fark nedir?

Makine öğrenmesine köprü, veri etiği ve KVKK

Bu son modül iki konuyu birleştirir. İlki, veri analizinden makine öğrenmesine nasıl geçildiği ve bir analistin bu geçişte hangi becerileri taşıdığıdır. İkincisi, veriyle çalışan herkesin bilmesi gereken etik ilkeler ve Türkiye'de kişisel verileri koruyan 6698 sayılı Kişisel Verilerin Korunması Kanunu'dur.

Analizden makine öğrenmesine

Analist geçmiş veriyi özetler ve açıklar. Makine öğrenmesi ise geçmiş veriden bir örüntü öğrenip yeni durumlar için tahmin üretir. İkisi arasında keskin bir çizgi yoktur; dördüncü modüldeki istatistik, üçüncü modüldeki temizlik ve altıncı modüldeki SQL, makine öğrenmesi projelerinin de temelidir.

Makine öğrenmesinde kurallar elle yazılmaz. Sisteme çok sayıda örnek verilir ve sistem girdilerle sonuç arasındaki ilişkiyi kendisi bulur. Temel kavramlar:

- **Özellik (feature):** Modelin kullandığı girdi değişkenleri. Müşterinin sipariş sayısı, son siparişinden bu yana geçen gün, şikâyet sayısı.
- **Etiket (label):** Tahmin edilmek istenen sonuç. Müşteri ayrıldı mı, ayrılma mı?
- **Eğitim verisi ve test verisi:** Model, verinin bir kısmıyla eğitilir; ayrılan diğer kısımla, daha önce görmediği veride ne kadar başarılı olduğu ölçülür.
- **Aşırı öğrenme (overfitting):** Modelin eğitim verisini ezberleyip yeni veride başarısız olması.

Makine öğrenmesi problem türleri

- **Regresyon:** Sayısal bir değer tahmin etmek. Gelecek haftanın talebi, bir evin fiyatı, bir siparişin teslim süresi.
- **Sınıflandırma:** Bir kategori tahmin etmek. Müşteri ayrılacak mı, işlem dolandırıcılık mı, e-posta istenmeyen ileti mi?
- **Kümeleme:** Etiket olmadan benzer kayıtları gruplamak. Müşterileri satın alma davranışına göre segmentlere ayırmak.
- **Anomali tespiti:** Olağandışı kayıtları bulmak. Beklenmedik bir fatura tutarı, bir makinedeki olağandışı titreşim.
- **Öneri:** Kullanıcıya ilgisini çekebilecek ürün ya da içerik önermek.

İlk ikisi etiketli veriyle yapılan **denetimli öğrenme**, kümeleme ise etiketsiz veriyle yapılan **denetimsiz öğrenme** örneğidir.

Kısa bir sınıflandırma örneği

Python'daki scikit-learn kütüphanesiyle, geçmiş müşteri verisinden ayrılma olasılığını tahmin eden basit bir model şöyle kurulur:

```
import pandas as pd
from sklearn.model_selection import train_test_split
from sklearn.linear_model import LogisticRegression
from sklearn.metrics import accuracy_score

# Geçmiş müşteri verisi; "ayrıldı" sütunu etiket (1 = ayrıldı, 0 = kaldı)
df = pd.read_csv("musteriler.csv")
X = df[["siparis_sayisi", "son_siparisten_gun", "sikayet_sayisi"]]
y = df["ayrildi"]

# Verinin yüzde 20'sini test için ayır, modeli kalan kısımla eğit
X_egitim, X_test, y_egitim, y_test = train_test_split(X, y, test_size=0.2,
random_state=42)
model = LogisticRegression().fit(X_egitim, y_egitim)
print("Test doğruluğu:", accuracy_score(y_test, model.predict(X_test)))
```

Bu örnek kavramı göstermek içindir. Gerçek bir projede doğruluk tek başına yeterli değildir. Müşterilerin yalnız küçük bir kısmı ayrılıyorsa, "kimse ayrılmayacak" diyen bir model de yüksek doğruluk alır. Bu yüzden kesinlik ve duyarlılık gibi ölçülere de bakılır ve model, basit bir **karşılaştırma noktasıyla** kıyaslanır. Örneğin talep tahmininde "geçen yılın aynı haftası" kadar basit bir yöntemi geçemeyen model kullanılmaya değmez.

Analistin makine öğrenmesine taşıdığı beceriler

- İş sorusunu doğru tanımlamak: Ne tahmin edilecek ve tahmin hangi kararı değiştirecek?
- Veriyi temizlemek ve anlamak: Bir projede emeğin büyük kısmı hâlâ buradadır.
- Özellik üretmek: "Son 90 gündeki sipariş sayısı" gibi anlamlı değişkenler türetmek, iş bilgisi gerektirir.
- Sonucu yorumlamak ve anlatmak: Modelin ne zaman yanıldığını ve sınırlarını açıkça söylemek.

Üretken yapay zekâ araçları da analiz işinde yardımcı olarak kullanılabilir: formül ya da sorgu taslağı önermek, bir kodu açıklamak gibi. Ama bu araçlar yanlış formülü de kendinden emin bir dille önerebilir. Üretilen her sorgu ve sonuç kaynak veriyle doğrulanır; kişisel veri ve şirket verisi, kurumun onaylamadığı araçlara yapııştırılmaz.

Veri etiği

Yasal olan her şey etik olmayabilir. Veriyle çalışırken şu ilkeler yol gösterir:

- **Amaç sınırlılığı:** Veri, toplandığı amaçla uyumlu biçimde kullanılır. Kargo takibi için alınan telefon numarası, habersizce başka bir pazarlama işine aktarılmaz.
- **Veri en azaltımı:** Yalnız gerekli olan veri toplanır ve saklanır.
- **Adalet ve önyargı:** Model geçmiş verideki önyargıyı öğrenebilir. Geçmişte belirli bir gruba daha az kredi verilmişse, bu veriyle eğitilen model aynı dengesizliği sürdürebilir. Cinsiyet ya da yaş gibi alanları modelden çıkarmak tek başına yetmez; posta kodu gibi başka alanlar bu bilgileri dolaylı olarak taşıyabilir. Sonuçlar farklı gruplar için ayrıca kontrol edilir.
- **Şeffaflık:** İnsanlar hakkında verilen kararlarda, kararın hangi bilgilere dayandığı açıklanabilir olmalıdır.

- **Hesap verebilirlik:** Model bir öneri sunar; insanları önemli ölçüde etkileyen kararlarda son karar ve sorumluluk bir insandır.
- **Güvenlik:** Veri yetkisiz erişime karşı korunur; ihtiyaç kalmayınca silinir ya da anonim hale getirilir.

KVKK'nın temel kavramları

6698 sayılı Kişisel Verilerin Korunması Kanunu 2016'da yürürlüğe girdi ve uygulamasını Kişisel Verileri Koruma Kurumu denetler. Avrupa Birliği'ndeki kişilerin verisini işleyen kurumlar için AB Genel Veri Koruma Tüzüğü (GDPR) de gündeme gelebilir. Analistin bilmesi gereken temel kavramlar:

- **Kişisel veri:** Kimliği belirli ya da belirlenebilir bir gerçek kişiye ilişkin her türlü bilgi. Ad, telefon, e-posta, TC kimlik numarası, IP adresi, müşteri numarası, plaka.
- **Özel nitelikli kişisel veri:** Kanunda sayılan ve daha sıkı korunan veriler. Sağlık verileri, biyometrik ve genetik veriler, din ve mezhep, siyasi düşünce, sendika üyeliği, ceza mahkûmiyeti gibi.
- **Veri sorumlusu:** Verinin işleme amaçlarını ve araçlarını belirleyen kişi ya da kurum. Genellikle çalıştığı şirket.
- **Veri işleyen:** Veri sorumlusu adına veriyi işleyen kişi ya da kurum. Örneğin şirketin bordrosunu hazırlayan dış hizmet firması.
- **İlgili kişi:** Verisi işlenen gerçek kişi. Müşteri, çalışan, hasta.

Kanunun genel ilkelerine göre kişisel veriler hukuka ve dürüstlük kurallarına uygun, doğru ve gerektiğinde güncel, belirli, açık ve meşru amaçlarla, amaçla bağlantılı, sınırlı ve ölçülü biçimde işlenir ve yalnız gerekli süre kadar saklanır. Kanun, işleme için açık rıza dışında da şartlar sayar; sözleşmenin kurulması ya da ifası, hukuki yükümlülüğün yerine getirilmesi gibi. Veri sorumlusunun ilgili kişileri aydınlatma ve veri güvenliğini sağlama yükümlülüğü vardır; ilgili kişilerin de verilerine ilişkin bilgi isteme, düzeltme ve silme talep etme gibi hakları bulunur.

Pratikte analist için en önemli kural şudur: Bir analizde kişisel veri kullanmadan önce bu kullanımın kurumun belirlediği amaç ve kurallar içinde olduğundan emin ol. Emin değilsen kurumundaki veri koruma sorumlusuna ya da hukuk birimine sor.

Anonimleştirme ve takma adlandırma

Birçok analiz için kişilerin kim olduğunu bilmek gerekmez. Bu durumda veri anonimleştirilir.

- **Anonimleştirme:** Verinin, başka verilerle eşleştirilse bile hiçbir şekilde belirli bir kişiyle ilişkilendirilemeyecek hale getirilmesidir. Gerçekten anonim hale gelmiş veri artık kişisel veri sayılmaz.
- **Takma adlandırma (pseudonymization):** Ad ve kimlik numarası gibi alanların bir kodla değiştirilmesidir. Kodu gerçek kimlikle eşleştiren anahtar bir yerde durduğu için bu veri hâlâ kişisel veridir. Güvenliği artırır ama anonimleştirme değildir.

Yalnız adı ve kimlik numarasını silmek çoğu zaman yeterli olmaz. Doğum tarihi, ilçe ve cinsiyet gibi **dolaylı tanımlayıcılar** bir araya geldiğinde küçük gruplarda kişiyi yeniden tanımlamak mümkün olabilir. Bu yüzden şu teknikler birlikte kullanılır:

1. **Doğrudan tanımlayıcıları çıkarmak:** Ad, TC kimlik numarası, telefon, e-posta, açık adres.
2. **Genelleştirmek:** Yaş yerine yaş grubu, açık adres yerine il, tam tarih yerine ay.
3. **Toplulaştırmak:** Tek tek kayıtlar yerine grup özetleri paylaşmak.

4. **Küçük grupları bastırmak:** Belirli bir sayının altında kişi içeren grupları göstermemek ya da birleştirmek. Her dolaylı tanımlayıcı birleşiminde en az k kişinin bulunmasını şart koşan yaklaşıma **k-anonimlik** denir.
5. **Maskelemek:** Değerin bir kısmını gizlemek. Telefonun yalnız son dört hanesini göstermek gibi.

```
import pandas as pd

df = pd.read_csv("hasta_kayitlari.csv")

# Doğrudan tanımlayıcıları çıkar
df = df.drop(columns=["ad_ soyad", "tc_kimlik_no", "telefon", "adres"])

# Yaşı 10 yıllık gruplara çevir (genelleştirme)
df["yas_grubu"] = (df["yas"] // 10 * 10).astype(str) + "-" + (df["yas"] // 10 * 10 + 9).astype(str)
df = df.drop(columns=["yas"])

# Aynı ilçe ve yaş grubunda 5'ten az kişi varsa bu satırları çıkar
grup_boyu = df.groupby(["ilce", "yas_grubu"])["ilce"].transform("size")
df = df[grup_boyu >= 5]
```

Bu kod yalnız temel adımları gösterir. Gerçek bir paylaşımda hangi alanların dolaylı tanımlayıcı olduğu, verinin kiminle paylaşılacağı ve yeniden tanımlama riski ayrıca değerlendirilir.

Uygulama

Bir şehir hastanesinin kalite birimi, acil servis bekleme sürelerini bir üniversitedeki araştırma ekibiyle paylaşmak istiyor. Veri setinde hasta adı, TC kimlik numarası, doğum tarihi, ikamet ilçesi, geliş tarihi ve saati, triyaj kategorisi, tanı kodu, bekleme süresi ve taburcu durumu var. Sağlık verisi özel nitelikli kişisel veridir; bu yüzden özen gerekir.

1. Paylaşımın amacını yaz: bekleme sürelerinin gün, saat ve triyaj kategorisine göre analizi. Bu amaç için hastanın kimliği gerekmiyor.
2. Hastanenin veri koruma sorumlusu ve hukuk birimiyle paylaşımın dayanağını ve koşullarını netleştir.
3. Ad, TC kimlik numarası ve diğer doğrudan tanımlayıcıları çıkar.
4. Doğum tarihini 10 yıllık yaş grubuna çevir.
5. Geliş zamanını saat dilimine genelleştir; tam dakika amaç için gerekmiyor.
6. Tanı kodu araştırma için gerekmiyorsa çıkar. Gerekiyorsa daha üst bir tanı grubuna genelleştir.
7. İlçe, yaş grubu ve tanı grubu birleşiminde az sayıda hasta kalan satırları bastır ya da grupları birleştir.
8. Sonucu, kişinin yeniden tanımlanıp tanımlanamayacağı açısından ikinci bir kişiye kontrol ettir.
9. Paylaşımı yalnız gerekli kişilerle, güvenli bir kanal üzerinden yap ve hangi verinin kime, ne zaman verildiğini kayıt altına al.

Sık yapılan hatalar

- Yalnız ad ve kimlik numarasını silip veriyi anonim saymak.
- Takma adlandırılmış veriyi anonim veri gibi serbestçe paylaşmak.
- Bir amaçla toplanan veriyi kimseye sormadan başka bir amaçla kullanmak.
- "Belki ileride lazım olur" diye gereğinden fazla kişisel veri toplamak ve süresiz saklamak.
- Modelden hassas alanları çıkarınca önyargının da ortadan kalktığını sanmak.
- Model doğruluğunu tek ölçüyle ve basit bir karşılaştırma yapmadan değerlendirmek.
- Kişisel veriyi kurumun onaylamadığı bir yapay zekâ aracına ya da kişisel bulut hesabına yüklemek.

Kendini dene

1. Regresyon ile sınıflandırma arasındaki fark nedir? Birer örnek ver.
2. Bir modeli neden basit bir karşılaştırma noktasıyla kıyaslamak gerekir?
3. Anonimleştirme ile takma adlandırma arasındaki fark nedir?
4. Sağlık verisi KVKK'ya göre hangi veri kategorisindedir?
5. Ad ve kimlik numarasını silmek neden her zaman anonimleştirme için yeterli değildir?

Kendini dene cevapları

Modül 1: 1) Ham veri tek bir kayıttır ("12 Mart, Şube 4, 540 TL"). Enformasyon düzenlenmiş özetir ("Şube 4'ün mart cirosu düştü"). İçgörü nedeni ya da örüntüyü söyler ("düşüş yol çalışmasıyla başladı"). Karar bu içgörüye dayanan adımdır ("bu dönemde ücretsiz teslimat yapalım"). 2) Tahmine dayalı analitik. 3) Örneğin: "Bu yılın ilk çeyreğinde Ege bölgesindeki ciro, geçen yılın aynı dönemine göre hangi ürün gruplarında ve hangi şehirlerde düştü? Sonuç ikinci çeyrek kampanya bütçesinin dağılımında kullanılacak." 4) Veriyi bulma, inceleme ve temizleme (veri hazırlama) adımına. 5) Sipariş sayısı da arttığı için mutlak iade sayısındaki artış yanıltıcıdır; oran, sipariş hacmindeki değişimi hesaba katar.

Modül 2: 1) Yapılandırılmamış veri. 2) Posta kodu bir miktar değil, bir etikettir (nitel, sınıflama ölçeği); sayılardan oluşması onu sayısal yapmaz. 3) Sıralama (ordinal) ölçeği. 4) Santigratta sıfır derece "sıcaklığın yokluğu" değildir; gerçek bir sıfır noktası olmadığı için oranlar anlamlı değildir. 5) Birincil veri amaca tam uyar ama toplaması zaman ve maliyet ister. İkincil veri hızlı ve ucuzdur ama tanımları ve toplanma biçimi ihtiyaca tam uymayabilir.

Modül 3: 1) Doğruluk, eksiksizlik, tutarlılık, geçerlilik, benzersizlik, güncellik (herhangi dördü). 2) Aynı şehrin farklı yazılışlarla ("İstanbul", "İst.", "İSTANBUL") girilmesini ve buna bağlı yanlış gruplamayı önler. 3) Boş değer "satış yok" anlamına gelmeyebilir (ör. sistem çalışmamış olabilir); sıfırla doldurmak ortalamayı aşağı çeker ve yanlış tablo verir. 4) Silmeden önce nedenini araştırırım: veri giriş hatası mı, gerçek bir kurumsal sipariş mi? Hataysa düzeltirim, gerçekse analizde ayrıca belirtirim. 5) Raporun güvenilirliğini ve tekrarlanabilirliğini artırır; sonuçlar değiştiğinde nedeninin izlenmesini ve başkasının aynı işi yapabilmesini sağlar.

Modül 4: 1) 4.200 TL'lik tek bir uç değer ortalamayı yaklaşık 786 TL'ye çeker; medyan (220 TL) uç değerden etkilenmez ve tipik siparişi daha iyi temsil eder. 2) Standart sapma (ya da çeyrekler arası açıklık gibi bir yayılım ölçüsü). 3) 1 yüzde puan artış; yüzde 25 artış. 4) Karıştırıcı değişken (bu örnekte sıcak hava). 5) Hayır. Örnekleme büyütme rastgele hatayı azaltır ama örneklemin seçilme biçimindeki sistematik sorunu düzeltmez; aynı yanlış sonuç daha büyük örnekleme de çıkar.

Modül 5: 1) Aynı değişken (ay) birden fazla sütuna dağılmış olur; filtrelemek, gruplamak, pivot tablo kurmak ve yeni ay eklemek zorlaşır. Doğrusu bir "Ay" ve bir "Tutar" sütunudur. 2) E1 hücresine mutlak başvurudur; formül kopyalandığında kaymaz. KDV oranı gibi tek hücrede tutulan sabit değerlere başvururken kullanılır. 3) ÇOKETOPLA (SUMIFS). 4) Satırlar, sütunlar, değerler, filtreler. 5) Fazladan boşluk, sayı ve metin olarak saklanan kodların eşleşmemesi, aranan değer diğer tabloda gerçekten bulunmaması.

Modül 6: 1) Birincil anahtar bir tablodaki her satırı benzersiz tanımlar. Yabancı anahtar, başka bir tablonun birincil anahtarına işaret ederek iki tabloyu ilişkilendirir. 2) WHERE gruplamadan önce satırları, HAVING gruplamadan sonra grupları süzer. 3) KapanisTarihi IS NULL. 4) LEFT JOIN (müşteri tablosu solda) ve sipariş tarafının boş olduğu satırları süzmek. 5) Sorgu mantıksal olarak WHERE'i SELECT'ten önce işler; WHERE çalışırken takma ad henüz oluşmamıştır.

Modül 7: 1) Çizgi grafik. 2) Az sayıda parça (yaklaşık iki ile beş arası) varsa ve parçaların toplamı gerçekten bir bütünü (yüzde 100) oluşturuyorsa. 3) Çubuğun boyu değeri temsil eder; eksen sıfırdan başlamazsa boylar orantısız olur ve küçük farklar büyük görünür. 4) Histogram (grupları karşılaştırmak için kutu grafik). 5) Örneğin: "Ege bölgesinde satışlar üç aydır düşüyor, diğer bölgeler hedefte."

Modül 8: 1) Aynı göstergenin farklı birimlerde farklı hesaplanmasını ve toplantılarda farklı rakamların tartışılmasını önler. 2) Olgu tablosu ölçülen olayları (satış, sevkiyat) ve sayısal değerleri tutar. Boyut tabloları bu olayları tanımlayan bilgileri (tarih, müşteri, ürün) tutar ve filtreleme ile gruplamada kullanılır. 3) Her ölçüm bir metriktir; KPI, bir hedefe bağlı, sayıca az ve özenle seçilmiş, performansı izlemek için kullanılan metriktir. 4) Öncü: satış ekibinin haftalık müşteri ziyareti sayısı. Gecikmeli: aylık ciro. 5) Ad, amaç, formül, kapsam (dahil ve hariç olanlar), veri kaynağı, güncelleme sıklığı, hedef ve eşikler, sorumlu.

Modül 9: 1) Eşzamanlı erişim, bütünlük kuralları, yetkilendirme, yedekleme ve geri yükleme, performans (herhangi üçü). 2) Var olmayan bir müşteriye bağlı sipariş girilmesini, yani tablolar arasındaki bağlantının bozulmasını önler. 3) Bölünmezlik (Atomicity), tutarlılık (Consistency), yalıtım (Isolation), kalıcılık (Durability). 4) OLTP günlük işlemleri kaydeden, çok sayıda küçük ve hızlı işlem yapan sistemdir. OLAP analiz için tasarlanmış, çok satırı tarayan büyük sorgular çalıştıran sistemdir. 5) Veri ambarında veri temizlenmiş, birleştirilmiş ve analiz için düzenlenmiştir. Veri gölünde her türden veri ham biçimde saklanır.

Modül 10: 1) Regresyon sayısal bir değer tahmin eder (gelecek haftanın talebi). Sınıflandırma bir kategori tahmin eder (müşteri ayrılacak mı, ayrılmayacak mı). 2) Basit bir yöntemi geçemeyen modelin karmaşıklığına ve maliyetine değmez; yüksek doğruluk gibi görünen sonuçlar da ancak bir karşılaştırma noktasına göre anlam kazanır. 3) Anonim veri hiçbir şekilde bir kişiyle ilişkilendirilemez ve kişisel veri sayılmaz. Takma adlandırılmış veride kimlik bir kodla değiştirilmiştir; anahtarla geri dönülebildiği için hâlâ kişisel veridir. 4) Özel nitelikli kişisel veri. 5) Doğum tarihi, ilçe, cinsiyet gibi dolaylı tanımlayıcılar bir araya geldiğinde, özellikle küçük gruplarda, kişi yeniden tanımlanabilir.

Sözlük

A/B testi: Kullanıcıların rastgele iki gruba ayrılıp farklı sürümleri gördüğü ve sonuçların karşılaştırıldığı kontrollü deney.

ACID: Veri tabanı işlemlerinin bölünmezlik, tutarlılık, yalıtım ve kalıcılık özelliklerinin baş harfleri.

Anakütle (popülasyon): İncelenmek istenen grubun tamamı.

Anonimleştirme: Verinin hiçbir şekilde belirli bir kişiyle ilişkilendirilemeyecek hale getirilmesi.

Aykırı değer: Verinin genelinden belirgin biçimde farklı olan değer.

Betimleyici analitik: "Ne oldu?" sorusuna cevap veren analiz.

Birincil anahtar (primary key): Bir tablodaki her satırı benzersiz biçimde tanımlayan sütun.

Boyut tablosu: Olayları tanımlayan tarih, müşteri, ürün gibi bilgileri tutan tablo.

Büyük veri: Hacmi, hızı ya da çeşitliliği nedeniyle geleneksel araçlarla işlenemeyen veri.

Çeyrekler arası açıklık (IQR): Üçüncü çeyrek ile birinci çeyrek arasındaki fark.

DAX: Power BI'da ölçü ve hesaplanmış sütun yazmak için kullanılan formül dili.

Dizin (index): Sık sorgulanan sütunlarda aramayı hızlandıran veri tabanı yapısı.

Düzenli veri (tidy data): Her satırın bir kaydı, her sütunun bir değişkeni taşıdığı tablo yapısı.

ETL: Veriyi kaynaktan çekme, dönüştürme ve hedefe yükleme süreci.

Gösterge paneli (dashboard): Önemli göstergeleri tek ekranda, genellikle etkileşimli olarak sunan rapor.

Hata payı: Örneklemden hesaplanan bir sonucun belirsizliğini gösteren aralık.

İş zekâsı (BI): Kurum verisini toplayıp düzenleyen ve rapor ile panellerle karar vericilere sunan araç ve süreçler.

JOIN: SQL'de iki tabloyu ortak bir sütun üzerinden birleştiren işlem.

Karıştırıcı değişken: İki değişkeni birden etkileyerek aralarında yanıltıcı bir ilişki oluşturan üçüncü etken.

Korelasyon: İki değişkenin birlikte değişme eğilimi; -1 ile +1 arasında ölçülür.

KPI (temel performans göstergesi): Bir hedefe ne kadar yaklaşıldığını gösteren, özenle seçilmiş metrik.

KVKK: 6698 sayılı Kişisel Verilerin Korunması Kanunu.

Medyan (ortanca): Sıralanmış verinin tam ortasındaki değer.

Normalizasyon: Tekrarı önlemek için veriyi ilişkili tablolara bölme işlemi.

NoSQL: Tablo dışındaki veri modellerini (doküman, anahtar-değer, geniş sütunlu, çizge) kullanan veri tabanlarının genel adı.

OLAP: Analiz ve raporlama için tasarlanmış veri sistemleri.

OLTP: Günlük işlemleri kaydeden veri sistemleri.

Örneklem: Anakütleden seçilip gerçekten incelenen alt grup.

Özel nitelikli kişisel veri: Sağlık, biyometrik ve genetik veri gibi KVKK'da sayılan ve daha sıkı korunan veriler.

Pivot tablo: Bir veri tablosunu sürükle-bırak yöntemiyle özetleyen elektronik tablo aracı.

Power Query: Excel ve Power BI'da veriyi alma, temizleme ve dönüştürme aracı.

SQL: İlişkisel veri tabanlarını sorgulamak ve yönetmek için kullanılan standart dil.

Standart sapma: Değerlerin ortalamadan tipik olarak ne kadar uzaklaştığını gösteren yayılım ölçüsü.

Takma adlandırma (pseudonymization): Kimlik bilgilerinin bir kodla değiştirilmesi; anahtarla geri dönülebildiği için veri kişisel veri olmaya devam eder.

Veri ambarı (data warehouse): Farklı kaynaklardan gelen verinin temizlenip analiz için düzenlendiği merkezî depo.

Veri gölü (data lake): Her türden verinin ham biçimde saklandığı depo.

Veri sözlüğü: Bir veri setindeki alanların anlamını, türünü, birimini ve izin verilen değerlerini anlatan belge.

Yabancı anahtar (foreign key): Başka bir tablonun birincil anahtarına işaret eden sütun.

Yıldız şeması: Ortada bir olgu tablosu ve çevresinde boyut tablolarından oluşan veri modeli.

Sonraki adım

Eğitimi bitirdiysen bilgini ölç. Daphne Kariyer'de bu eğitimin konularıyla ilgili şu testler ücretsizdir: Veri Bilimi, Veri Analizi, Data Analytics, SQL, Power BI, Tableau, Büyük Veri ve Veri Tabanı Yönetimi. Her test 20 sorudan oluşur. Önce Veri Analizi ve SQL testleriyle başlayıp, ardından ilgilendiğin araca ya da alana göre diğerlerine geçebilirsin.

55 ve üzeri puan alırsan isteğe bağlı olarak doğrulanabilir dijital sertifika alabilirsin. Düşük puan aldığın testlerde ilgili modüle dönüp "Kendini dene" sorularını yeniden çözmek iyi bir tekrar yoludur.

daphnekariyer.com/dijital-yetkinlikler/