Pandas ve Polars Hızlandırma Teknikleri Kıyaslaması

Nedir?

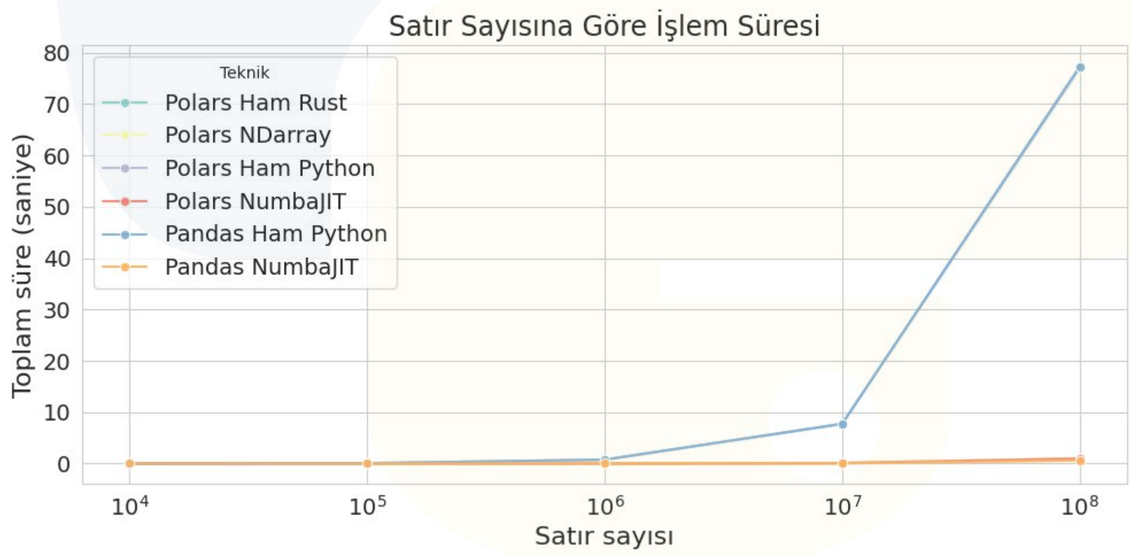
Veri işleme, günümüzde birçok alanda ihtiyaç haline geldi. Bunun için en yaygın araçlardan birisi de kuşkusuz Python dilinde Pandas ve Polars kütüphaneleri. Bu araçlar birçok dahili hızlandırma tekniği kullansa da veri üzerinde kompleks özel işlemler uygulamak gerektiğinde bu imkanlar daralıyor ve kullanıcı tanımlı fonksiyon (UDF) yazılması gerekiyor. Ancak, UDF'ler, özellikle gruplama yapılamayan koşullarda, paralelleştirme imkanları kısıtlandığı için artan veri ölçeğiyle ciddi performans sorunlarına dönüşebiliyor. Bu tür senaryolarda Python ile Numba JIT kullanma, ya da Rust'ta yeniden yazma gibi tekniklerle UDF'lerin hızlandırılması gündeme geliyor.

Peki, bu hızlandırma tekniklerine ne zaman ihtiyaç var ve kendi aralarında nasıl kıyaslanıyorlar?

Bu çalışmada; 10 bin ile 100 milyon satır aralığında, 3 sütundan oluşan, eksikli zaman serisi verilerini karakterize eden sentetik veriler üzerinde yapılan analizlere dayanarak yukarıdaki sorulara pratik yanıtlar verilmeye çalışılacaktır.

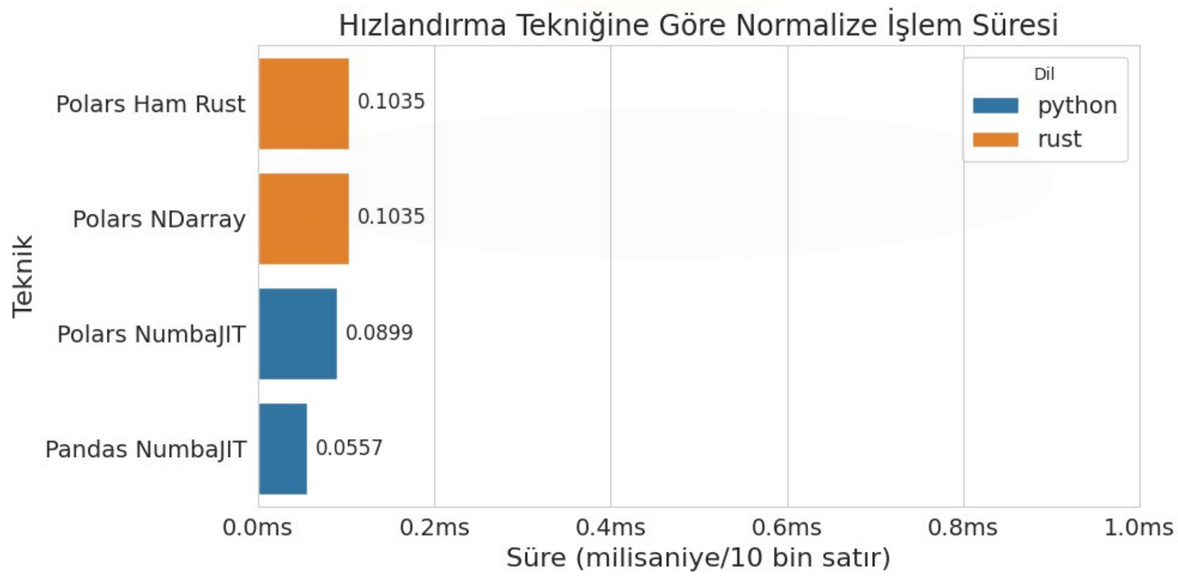
Hızlandırma ne zaman gerekli?

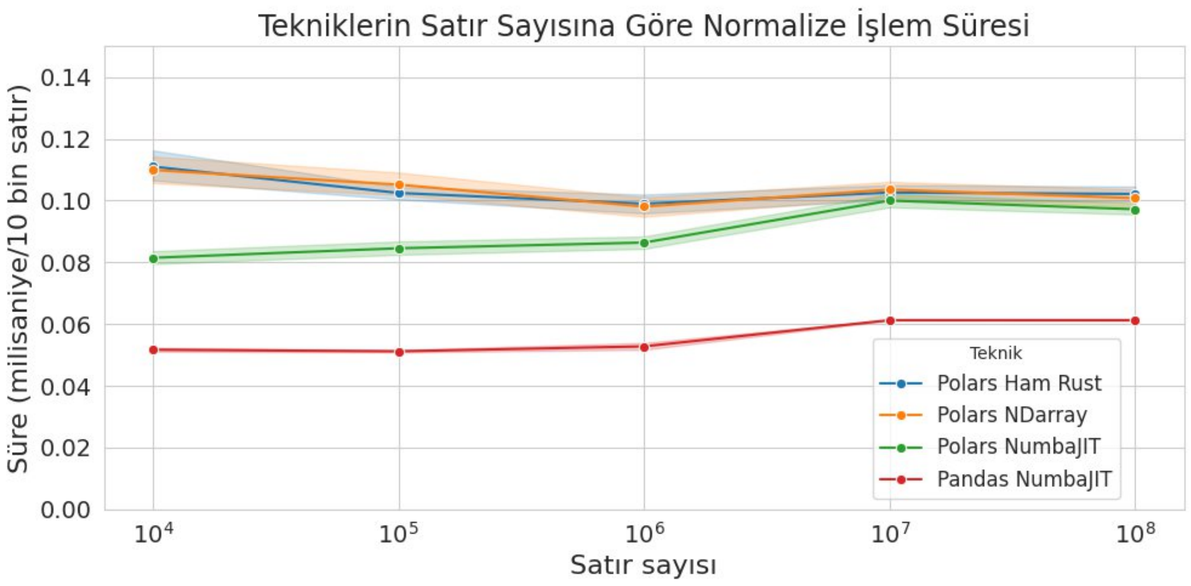
Verinin yapısı, ölçeği ve yapılan işlemlere göre bu sorunun yanıtı değişse de; yapılan çalışmaya göre, 1 milyon satıra kadar süre ölçeği yakın olduğu için hızlandırma tekniklerinin pratik etkisi düşük oluyor. **1 milyon satırdan sonra** ise veri ölçeğiyle beraber artan süreler göz önüne alındığında hızlandırma tekniklerinin etkisi daha belirgin hale geliyor.



Teknikler arasında fark var mı?

Fazla değil! Python'un yavaşlığı, hızlandırma tekniklerinden biri kullanıldığında ciddi anlamda azalıyor; Pandas ya da Polars'ı Numba JIT kullanarak hızlandırmak ile Rust'ta yeniden yazmak arasında istatistiksel olarak anlamlı fark olsa da zaman ölçeği nedeniyle pratik anlamda bir fark gözlemlemek güç, zira 10 bin satırın işlenmesi için gereken süre hala 1 milisaniyenin çok altında kalıyor.





Ya stabilite ve ölçeklenebilirlik?

Hızlandırma teknikleri, artan satır sayısı ve tekrarlı çalıştırmalarda stabil süreler sergiliyor, yani artan veriyle birlikte **stabil şekilde ölçeklendiklerini** söylemek mümkün.
En stabil teknik Pandas + Numba JIT olmakla birlikte, sürelerin ölçeği düşünüldüğünde pratikte anlamlı bir farktan bahsetmek zor.

ANADOLU ÜNİVERSİTESİ
Enes Selim Şenol
eselimsen@gmail.com

Kaynak (gizli depo, erişim için ulaşınız):
https://github.com/eselimsen/dpu500-data-imputation