

Veri analizi

husam salah

180401122

Python ile İstatistiksel Analiz: Veri Keşfi ve Hipotez Testleri

İstatistik, verilerin anlamını çıkarmak için güçlü bir araçtır. Python gibi programlama dilleri, istatistiksel analizler yapmak için harika birer araçtır. Bu blog yazısında, Python kullanarak popülasyonları oluşturma, örneklem seçme, istatistiksel testler yapma gibi işlemleri adım adım göreceğiz.

Popülasyon ve Örneklem Oluşturma:

Kod parçamız, rastgele normal dağılıma sahip popülasyonlar oluşturmak için `random.gauss()` fonksiyonunu kullanıyor. Örneklem büyüklüğü, ortalama ve standart sapma gibi parametreleri değiştirerek farklı popülasyonlar elde edebiliriz.

Herhangi bir veri setiyle çalışırken, genellikle örneklem çekmek ve bu örneklem üzerinden çeşitli analizler yapmak gerekebilir. Python'da, rastgele popülasyonlar oluşturmak ve bu popülasyondan örneklem seçmek oldukça kolaydır.

```
"""python
# Popülasyon ve örneklem oluşturma
population = get_population(10000, 50, 15)
sample = get_sample(100)
"""
```

Belirli bir popülasyondan örneklem seçmek için `random.choice()` yöntemini kullanıyoruz. Örneklem büyüklüğüne bağlı olarak farklı örneklem seçimleri yapabiliriz.

```
sample = get_sample(100)
```

Temel İstatistiksel Hesaplar:

Veri setlerinin anlaşılmasında ve açıklanmasında temel istatistiksel ölçümler önemlidir. Örneğin, ortalama ve standart sapma gibi değerler veri dağılımı hakkında fikir verir.

```
"""python
# Temel istatistiksel hesaplamalar
statistics = get_sample_statistics(sample)
"""
```

Hipotez Testleri: T-test ve Z-test:

Hipotez testleri, veri setlerinde bulunan farklılıkların tesadüfi olup olmadığını anlamak için kullanılır. Küçük örneklem için t-testi ve z-testi hesaplamaları bu farklılıkları değerlendirmemize yardımcı olur.

```
"""python
# Hipotez testleri: T-test ve Z-test
t_test_result = get_sample_statistics_from_small_sample(sample)
"""
```

P Değeri Hesaplama ve İstatistiksel Anlamlılık:

P değeri, veri setinde elde edilen farkların tesadüfi olma olasılığını ölçer. Bu değer, veri setindeki farklılıkların tesadüfi olup olmadığını belirlememize yardımcı olur.

```
"""python
# P değeri hesaplama
p_value = calculate_p_value(t_test_result)
"""
```

Grafiksel Analiz:

Veri görselleştirme, verilerin anlaşılmasını ve yorumlanmasını kolaylaştırır. Grafikler kullanarak elde edilen sonuçları görselleştirmek, veri analizinin daha net olmasını sağlar.

```
"""python
# Grafiksel analiz
plot_histogram(population)
plot_boxplot(sample)
"""
```

Bu adımların her biri, veri analizindeki önemli aşamaları temsil eder. İstatistiksel analizlerin nasıl yapıldığını anlamak için her bir adımın ayrıntılı olarak açıklanması ve kod örnekleriyle desteklenmesi önemlidir. Bu adımlar, veri analizi yapmak isteyenler için faydalı bir rehber olabilir.