

### 1. **\*\*Veri Setinin Anlaşılması ve Yüklmesi:\*\***

- Veri seti, projenin temel veri kaynağıdır ve projenin amacını belirleyen bilgileri içerir. Bu aşamada, veri setinin hangi formatlarda olduğunu (CSV, Excel, SQL) anlamalısınız. Örneğin, "insurance.csv" adlı bir CSV dosyası olabilir. Bu dosyayı `pd.read_csv()` fonksiyonu ile yükleyebilirsiniz.

### 2. **\*\*Veri Keşfi ve Analizi:\*\***

- Veri setini incelemek, veri hakkında ön bilgi edinmeniz için önemlidir. `df.head()` ile ilk birkaç satırı görüntüleyerek veri yapısını anlayabilirsiniz. `df.info()` ile veri türlerini, eksik verileri ve bellek kullanımını kontrol edebilirsiniz. `df.describe()` ile temel istatistikleri görüntüleyebilir, sayısal değişkenlerin dağılımlarını anlayabilirsiniz.

### 3. **\*\*Veri Ön İşleme:\*\***

- Kategorik veriler, model eğitimi için sayısal formatta olmalıdır. Bu nedenle, kategorik verileri sayısal hale dönüştürmelisiniz. Örneğin, `LabelEncoder` ile kategorik sütunları sayısal olarak temsil edebilirsiniz. Bu adım, modele girdi sağlamak için gereklidir. Ayrıca, bağımlı (hedef) ve bağımsız değişkenleri belirlemelisiniz.

### 4. **\*\*Veri Ölçeklendirme:\*\***

- Veri ölçeklendirme, özelliklerin aynı ölçeğe sahip olmasını sağlar. Bu, modelin özellikleri yanıltmadan daha iyi performans göstermesini sağlar. Örneğin, `Standard Scaling` ile özelliklerin ortalaması 0 ve standart sapması 1 olacak şekilde ölçeklendirilir. Bu, özelliklerin farklı büyüklüklerde olmasını engeller.

### 5. **\*\*Model Seçimi ve Eğitimi:\*\***

- Model, veriyi temsil eden matematiksel bir yapıdır. Başlangıç için basit bir model seçmek, projenin genel yönünü belirlemeye yardımcı olur. Örneğin, Lineer Regresyon basit bir modeldir. Modeli eğitirken, veri setini modele veri olarak verirsiniz. Model, veriyi analiz eder ve içerdği örüntüleri öğrenir.

### 6. **\*\*Model İyileştirme:\*\***

- Başlangıçta kullanılan model belirli bir performansa sahip olabilir. Daha karmaşık modeller, daha fazla öğrenme kapasitesine sahip olabilir ve daha iyi sonuçlar verebilir. Bu nedenle, başlangıç modelinden daha karmaşık modellere geçebilirsiniz. Ayrıca, hiperparametreler ile modelin davranışını ayarlayabilirsiniz. Örneğin, Rastgele Orman Regresyonu gibi modelleri deneyebilirsiniz.

### 7. **\*\*Model Değerlendirmesi:\*\***

- Model performansı, metriklerle ölçülür. MSE (Mean Squared Error), tahmin hatalarının karesel ortalama hatasını temsil eder. Düşük MSE, daha iyi tahmin anlamına gelir. MAE (Mean Absolute Error), tahmin hatalarının mutlak ortalama hatasını temsil eder. RMSE (Root Mean Squared Error), tahmin hatalarının karekökünün ortalama hatasını temsil eder. R2 skoru, modelin ne kadar iyi uyduğunu gösteren bir metriktir.

### 8. **\*\*Sonuçları Sunma:\*\***

- Projeyi paylaşmak ve sunmak önemlidir. GitHub'da bir depo oluşturarak projenizi kodlarla birlikte paylaşabilirsiniz. README dosyasında projenizin amacını, kullanılan adımları ve

sonuları aıklayan bir aıklama eklemelisiniz. Kodun her adımına yorum eklemek, projenin anlařılabilirlięini artırır ve bařkalarının projenizi incelemesini kolaylařtırır.