

Çanakkale Onsekiz Mart Üniversitesi Mühendislik Fakültesi Bilgisayar Mühendisliği Bölümü

BLM3012 - İLERİ VERİTABANI SİSTEMLERİ DERSİ FİNAL PROJESİ RAPORU

Baha YOLAL – 200401031 İrfan KARA – 210401075 Ali İhsan REY – 210401115

Öğr. Gör. İSMAİL KAHRAMAN

Bu raporda, bir Volkswagen araba fiyat tahmini uygulaması geliştirilmiş olup, veriler ilişkisel veri tabanı kullanılarak depolanmıştır. Veri tabanı işlemlerini kolaylaştırmak için ORM yapısı oluşturulmuş ve XAF kullanılarak veriler web ortamında gözlemlenmiştir. Aşağıda verilen aşamalar tek tek izlenmiş ve kontroller sağlanmıştır. (Yapılan aşamalar en altta verilen youtube linkinde özetlenmiştir.)

1) Data Mining Project

Volkswagen Araba Fiyat Tahmini:

Açıklama:

• Araba markası, modeli, yaş, kilometre, motor özellikleri gibi özellikler kullanılarak bir regresyon modeli geliştirilir ve araç fiyatları tahmin edilir.

Dataset Shape:

- Veri kümesinde toplam 15157 örnek (instance) bulunmaktadır.
- Her bir örnek 9 özellik (feature) içermektedir.

Feature Türleri:

Marka (Categoric), Model (Categoric), Yaş (Numeric), Kilometre (Numeric) Vites tipi (Categoric), Yakıt tipi (Categoric), Vergi (Numeric), Yakıt tüketimi (Numeric) Motor özellikleri (Numeric)

Kullanılan Yöntem:

- Kullanılan yöntem, bir bağımlı değişkenin sürekli bir değerini tahmin etmek için kullanılan bir gözetimli öğrenme (supervised learning) tekniği olan XGBoost yöntemidir.
- Bir çok gözetimli öğrenme tekniği denenmiş olup arasından XGBoost tekniği en iyi sonucu vermiştir. **XGBoost** (**Extreme Gradient Boosting**), gözetimli öğrenme algoritmalarından biridir ve özellikle yüksek doğruluk, hız ve verimlilik için optimize edilmiştir.
- 1) VwCar.csv dosyasındaki verilerin yapısına uygun olarak ORM (Object-Relational Mapping) yapısı oluşturulmuştur. Bu yapı, veri tabanı nesnelerini kod seviyesinde temsil eder ve veri tabanı işlemlerini kolaylaştırmak için yapılmıştır.

```
public virtual string model { get; set; } // 15157 non-null object

public virtual int? year { get; set; } // 15157 non-null int64

public virtual int? price { get; set; } // 15157 non-null int64

O basyuru
public virtual string transmission { get; set; } // 15157 non-null object

O basyuru
public virtual int? mileage { get; set; } // 15157 non-null int64

O basyuru
public virtual string fuelType { get; set; } // 15157 non-null object

O basyuru
public virtual int? tax { get; set; } // 15157 non-null int64

O basyuru
public virtual double? mpg { get; set; } // 15157 non-null float64

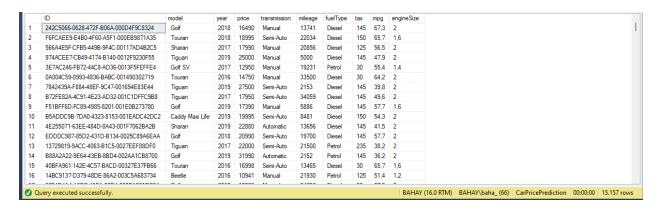
O basyuru
public virtual double? engineSize { get; set; } // 15157 non-null float64
```

2) Local db ye erişip 2.adımda oluşan db ye erişim sağlandı.

SQL	.Query1.sql - B(BAHAY\bah	a_ (66))* BahaY.	CarPricePrediction	on - dbo.VwCars	₽	×
	Column Name	Data Type	Allow Nulls			
₽¥		uniqueidentifier				
	model	nvarchar(MAX)	$\overline{\sim}$			
	year	int	\sim			
	price	int	$\overline{\smile}$			
	transmission	nvarchar(MAX)	$\overline{\mathbf{c}}$			
	mileage	int	$\overline{\smile}$			
	fuelType	nvarchar(MAX)	$\overline{\smile}$			
	tax	int	$\overline{\smile}$			
	mpg	float	$\overline{\mathbf{c}}$			
	engineSize	float	$\overline{\smile}$			

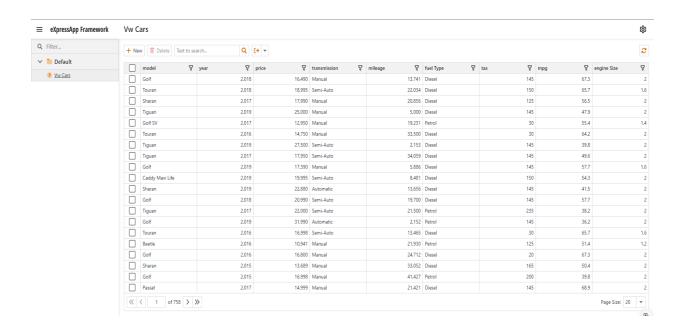
3) VwCar.csv dosyası, pandas DataFrame objesine aktarılmış ve ardından oluşan veriyi SQL ile veritabanına eklemek için INSERT komutu kullanılmıştır.

4) XAF' i run edip db ye yazılanların db de olduğundan eminiz ve artık arayüzde verileri görüyoruz.



6.) Test case yarattık ve bunu db' ye yazdık.

```
def predict_car_price(model, year, transmission, mileage, fuelType, tax, mpg, engineSize):
    # Veri cercevesi olusturma
    dataFrame = pd.DataFrame({
        "model: [model],
        "year: [year],
        "transmission: [transmission],
        "mileage: [mileage],
        "fuelType: [fuelType],
        "transmission: [transmission],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage: [mileage],
        "mileage],
        "mileage: [mileage],
        "mileage],
        F web projesi çalıştırılarak, oluşturulan verilerin ve test senaryolarının doğru bir şekilde işlendiği ve arayüzde görüntülendiği kontrol edilmiştir.



**Youtube Sunum Link:** 

https://youtu.be/ABoHSXg50P0

GitHub Proje Link:

https://github.com/Bahayll/VolkswagenCarPricePredictionMLApplication