Machine Learning Project

Project Topic: Diabetes Prediction

Term: Spring 2025

Prepared by:

Kerem Yıldırım

Project Description

In this project, we developed a diabetes prediction system using machine learning.
Our goal was to build a system that can accurately and effectively predict whether an individual has diabetes based on various health indicators such as glucose level, blood pressure, insulin level, age, and BMI.

By doing this, we aim to contribute to early diagnosis in healthcare and demonstrate a practical application of machine learning.

Dataset Introduction

We used the Pima Indian Diabetes dataset, which contains the health data of 768 individuals.

Although the majority of participants are female, the dataset is generalizable and usable for both genders.

Main independent variables:

- Number of pregnancies
- Glucose level
- Blood pressure
- Skin thickness
- Insulin
- Body Mass Index (BMI)
- Diabetes pedigree function
- Age

Dependent variable: Outcome (0: No diabetes, 1: Diabetes)

Note: Some variables have zero values (e.g., insulin = 0), which actually indicate missing data. These were addressed during model development.

Dataset Introduction

```
#Here we see 5 rows of the dataset
df = pd.read_csv("diabetes.csv")
df.head()
```

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

Class Distribution Chart

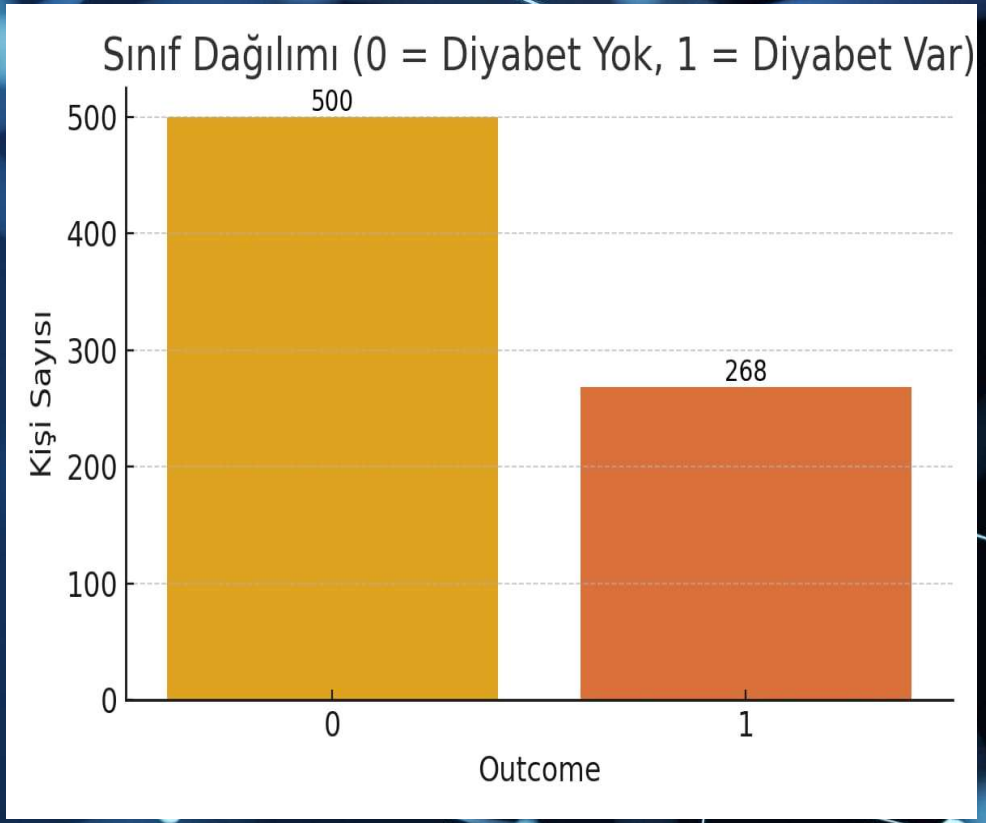
This chart shows the number of individuals with and without diabetes in the dataset.

Why is it important?

- This leads to class imbalance.
- The model tends to learn the majority class (0) better.
- Predicting the minority class (1) becomes more difficult.

Why do we use this chart?

- To see the impact of class balance (or imbalance) on model training.
- To determine whether methods like SMOTE or class weight adjustment are needed.



Correlation and Feature Analysis

A correlation analysis was conducted to understand the relationships between features in the dataset. It was observed that variables such as glucose, age, and BMI showed strong associations with diabetes.

This analysis made it possible to determine which variables the models should give more weight to.

```
# (We can see which features affect the output variable with correlation)
df.corr()
```

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
Pregnancies	1.000000	0.127911	0.208522	0.082989	0.056027	0.021565	-0.033523	0.544341	0.221898
Glucose	0.127911	1.000000	0.218367	0.192991	0.420157	0.230941	0.137060	0.266534	0.492928
BloodPressure	0.208522	0.218367	1.000000	0.192816	0.072517	0.281268	-0.002763	0.324595	0.166074
SkinThickness	0.082989	0.192991	0.192816	1.000000	0.158139	0.542398	0.100966	0.127872	0.215299
Insulin	0.056027	0.420157	0.072517	0.158139	1.000000	0.166586	0.098634	0.136734	0.214411
BMI	0.021565	0.230941	0.281268	0.542398	0.166586	1.000000	0.153400	0.025519	0.311924
DiabetesPedigreeFunction	-0.033523	0.137060	-0.002763	0.100966	0.098634	0.153400	1.000000	0.033561	0.173844
Age	0.544341	0.266534	0.324595	0.127872	0.136734	0.025519	0.033561	1.000000	0.238356
Outcome	0.221898	0.492928	0.166074	0.215299	0.214411	0.311924	0.173844	0.238356	1.000000

Correlation and Feature Analysis

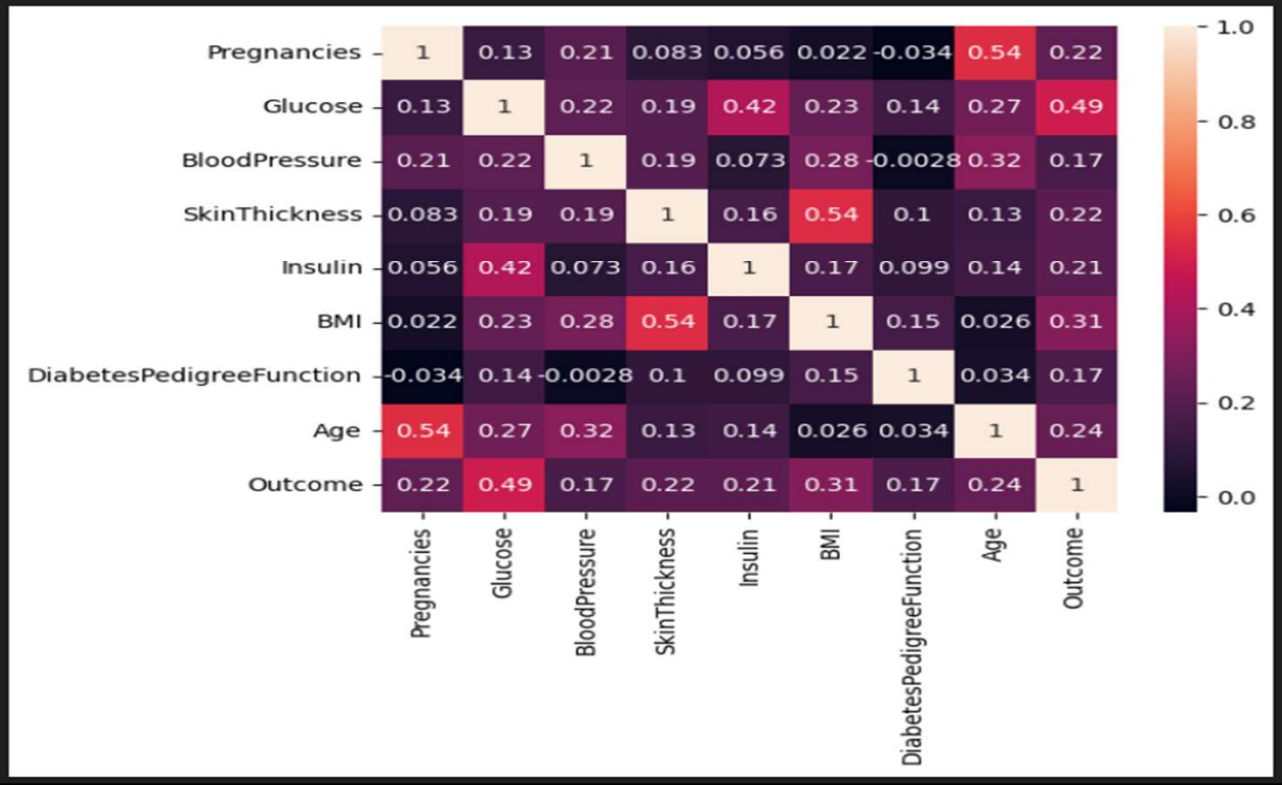
Why is the Correlation Matrix Important?

This chart shows the relationships between the variables in our dataset. It is especially used to understand how strongly the independent variables are related to the target variable, *Outcome* (diabetes: yes/no)..

```
#Correlation chart
import seaborn as sns

sns.heatmap(df.corr(), annot=True)
```

```
<Axes: >
```



Correlation and Feature Analysis

Which Variables Should Be Considered?

Glucose:
This is the variable most strongly associated with diabetes. As glucose levels increase, the risk of diabetes also increases.
➤ Therefore, it must be included in the model.

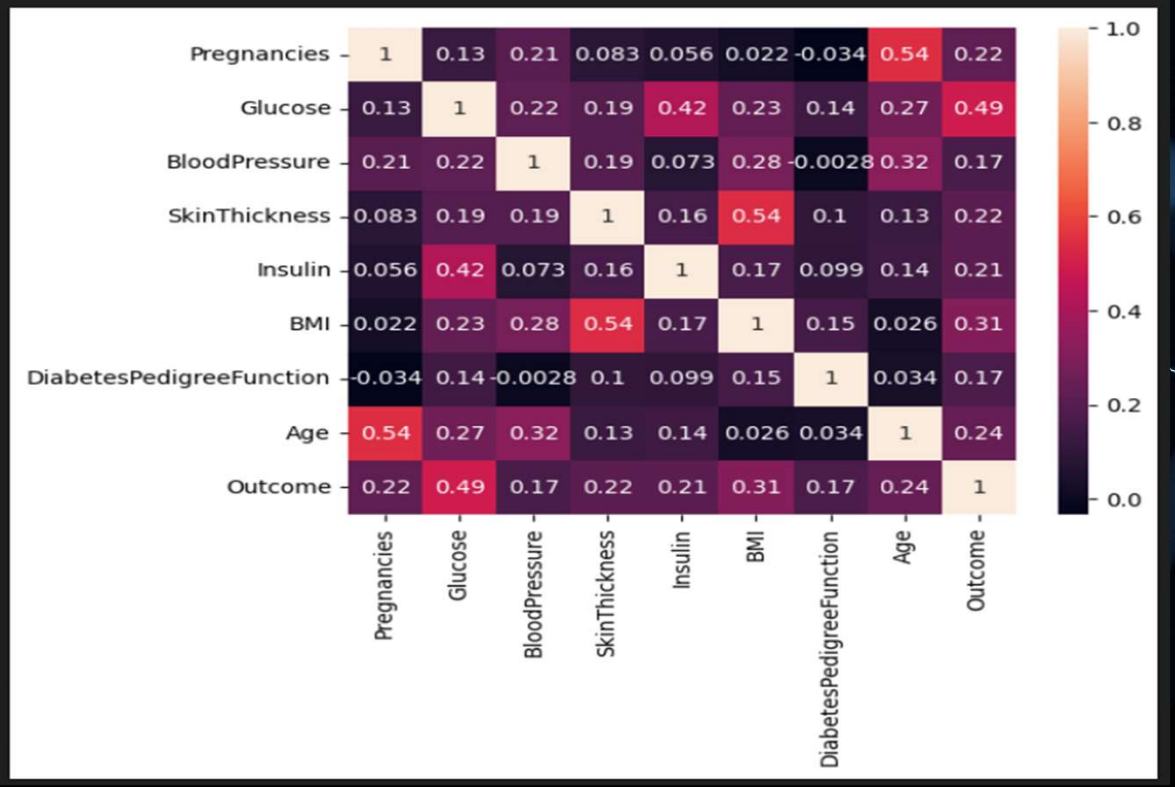
BMI (Body Mass Index):
Indicates whether a person is overweight. It is significantly associated with the risk of diabetes.
➤ It is an important risk indicator

Age:
As age increases, the risk of developing diabetes also rises.
➤ It should be taken into account in the model

```
#Correlation chart
import seaborn as sns

sns.heatmap(df.corr(), annot=True)
```

```
<Axes: >
```



Correlation and Feature Analysis

Why Do We Look at This Chart?

To determine which variables are meaningful for prediction and which are unnecessary or weak.

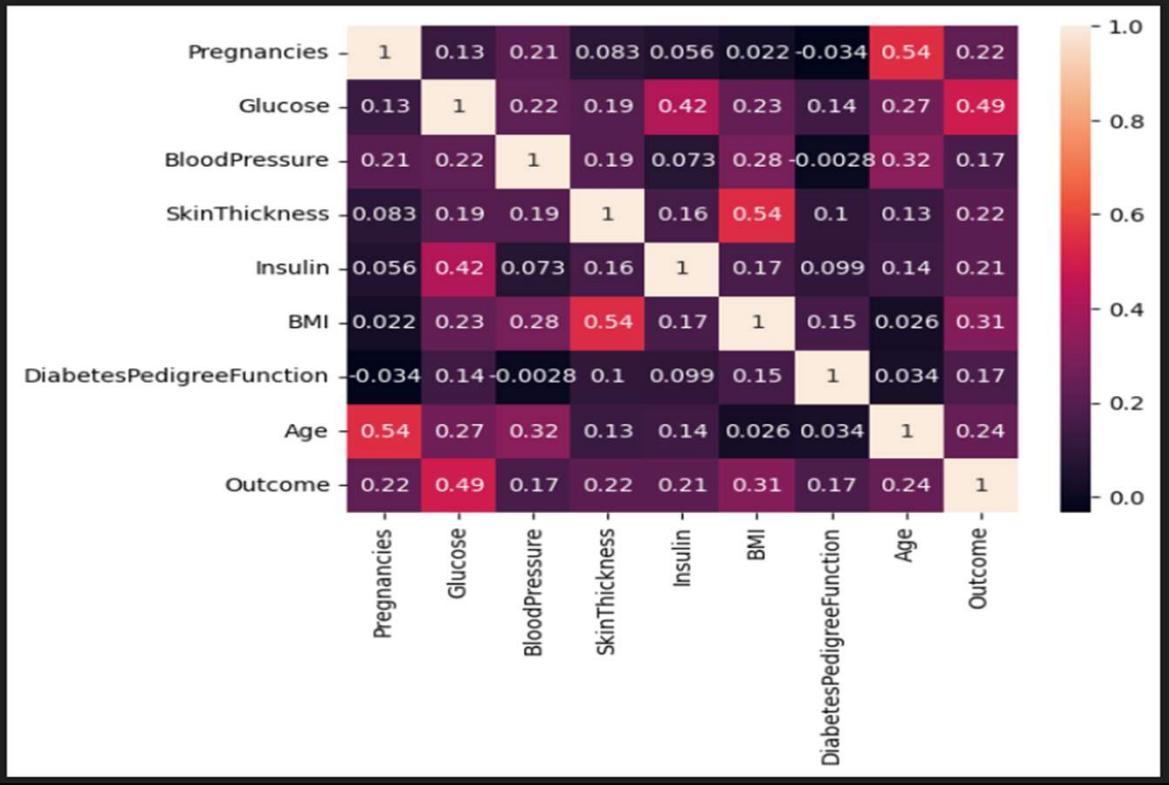
To select the right data and use the appropriate variables before building the model.

To avoid redundancy and improve model performance if there are variables with very high correlation.

```
#Correlation chart
import seaborn as sns

sns.heatmap(df.corr(), annot=True)
```

```
<Axes: >
```



Data Preprocessing

The following steps were applied before modeling:

Data Standardization:
All variables were transformed into a standard normal form to ensure the models operate more effectively.

Data Splitting:
The dataset was divided into 80% training and 20% testing.
These steps helped improve the model's generalizability and accuracy.

Machine Learning Models Used

Two different classification models were used in the project:

Logistic Regression

It is a simple, fast, and easy-to-understand algorithm.
It separates the data in a linear manner.
It is widely preferred in the healthcare field due to its high interpretability.

```
# Algoritmaların oluşturulması Lojistik Regression (Creating algorithms Logistic Regression)
from sklearn import linear_model
from sklearn.model_selection import cross_val_score
X = df[['Glucose','BMI','Age']]
y=df.iloc[:,8]
```

```
log_reg = linear_model.LogisticRegression()
log_reg_score = cross_val_score(log_reg,X,y,cv=10,scoring='accuracy').mean()
```

```
log_reg_score
```

```
np.float64(0.7669856459330144)
```

Machine Learning Models Used

Support Vector Machine (SVM) – Linear Kernel

It aims to find the optimal hyperplane that best separates the data.
A linear kernel was used to achieve linear separation.
Since it draws stricter boundaries, it has the potential for high accuracy.
A 10-fold cross-validation was applied to both models for robust evaluation.

```
#SVM Algoritmasını Çağıralım (Let's Call the SVM Algorithm)
from sklearn import svm
```

```
linear_svm = svm.SVC(kernel='linear')
```

```
linear_svm_score = cross_val_score(linear_svm,X,y,cv=10,scoring='accuracy').mean()
```

```
linear_svm_score
```

```
np.float64(0.7656527682843473)
```

Model Evaluation

The performance of the models was measured using various metrics. The ROC curve and AUC scores were particularly important indicators.

Logistic Regression AUC score: ~0.77

Similar AUC values were obtained with the SVM model.
While it is easier to predict non-diabetic individuals, correctly identifying diabetic patients is more challenging.

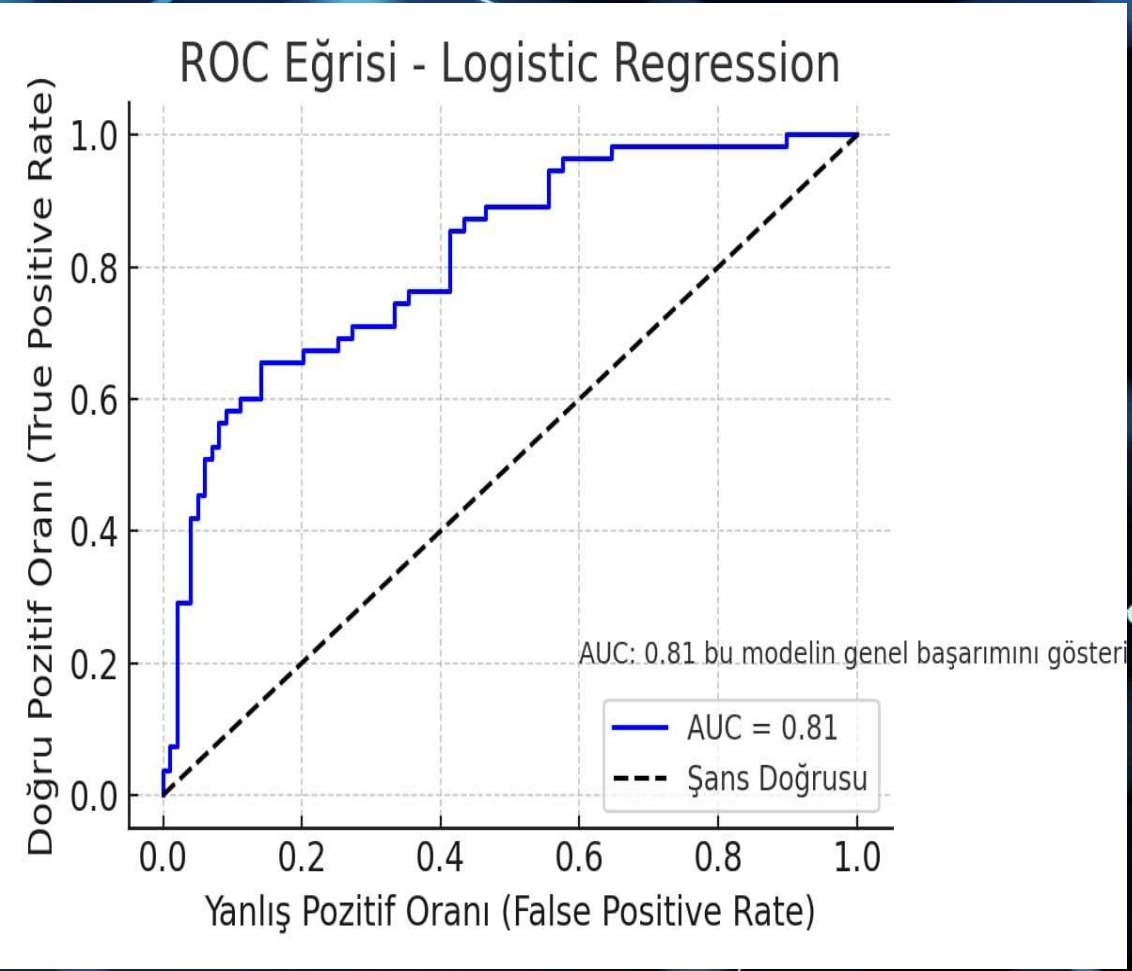
This is due to class imbalance in the dataset.

Model Evaluation – ROC Curve

Why is the ROC Curve Important?

The ROC curve shows how well our model performs classification.

It helps us evaluate the model's ability to make correct predictions at different threshold values.

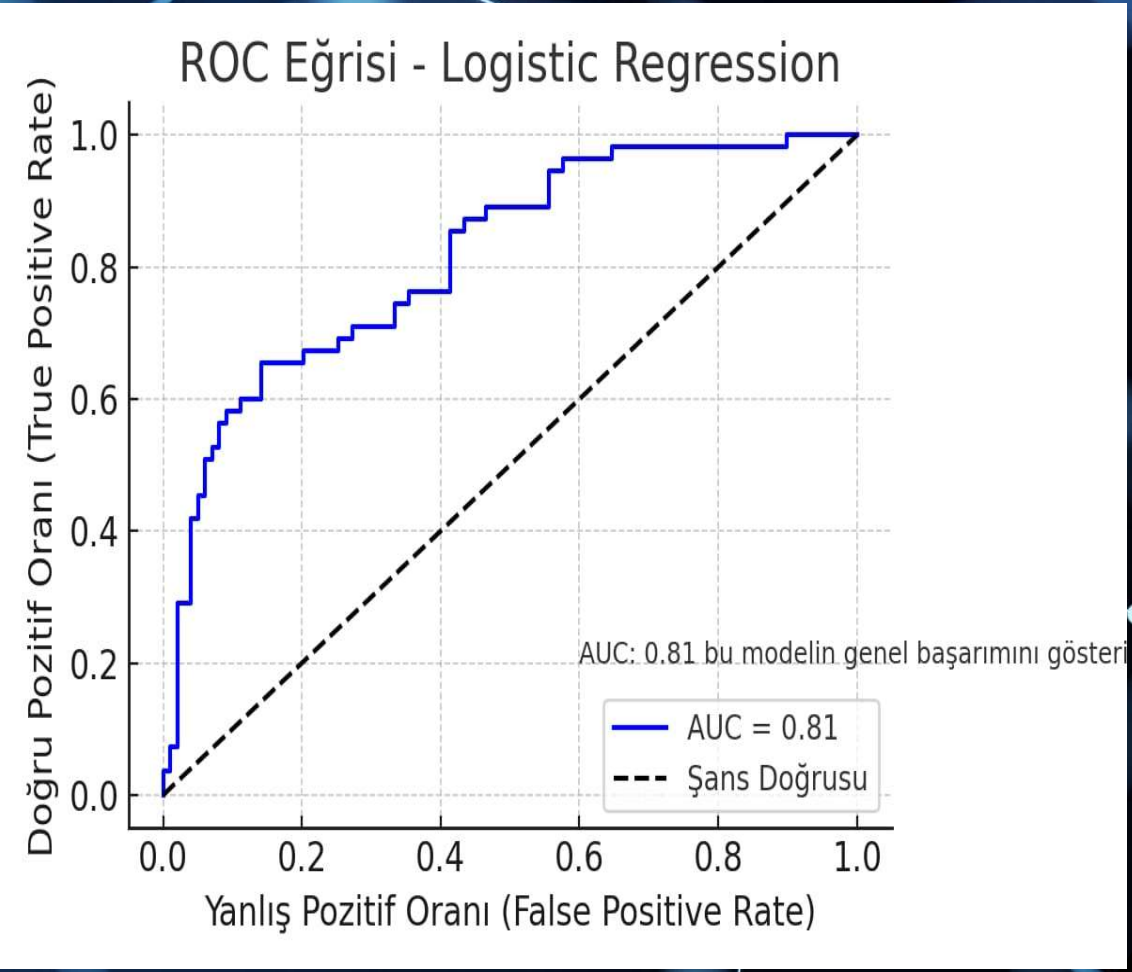


Model Evaluation – ROC Curve

What Does It Show?

Y-Axis (True Positive Rate):
The rate of correctly predicting individuals who actually have diabetes.

X-Axis (False Positive Rate):
The rate of incorrectly predicting non-diabetic individuals as diabetic.



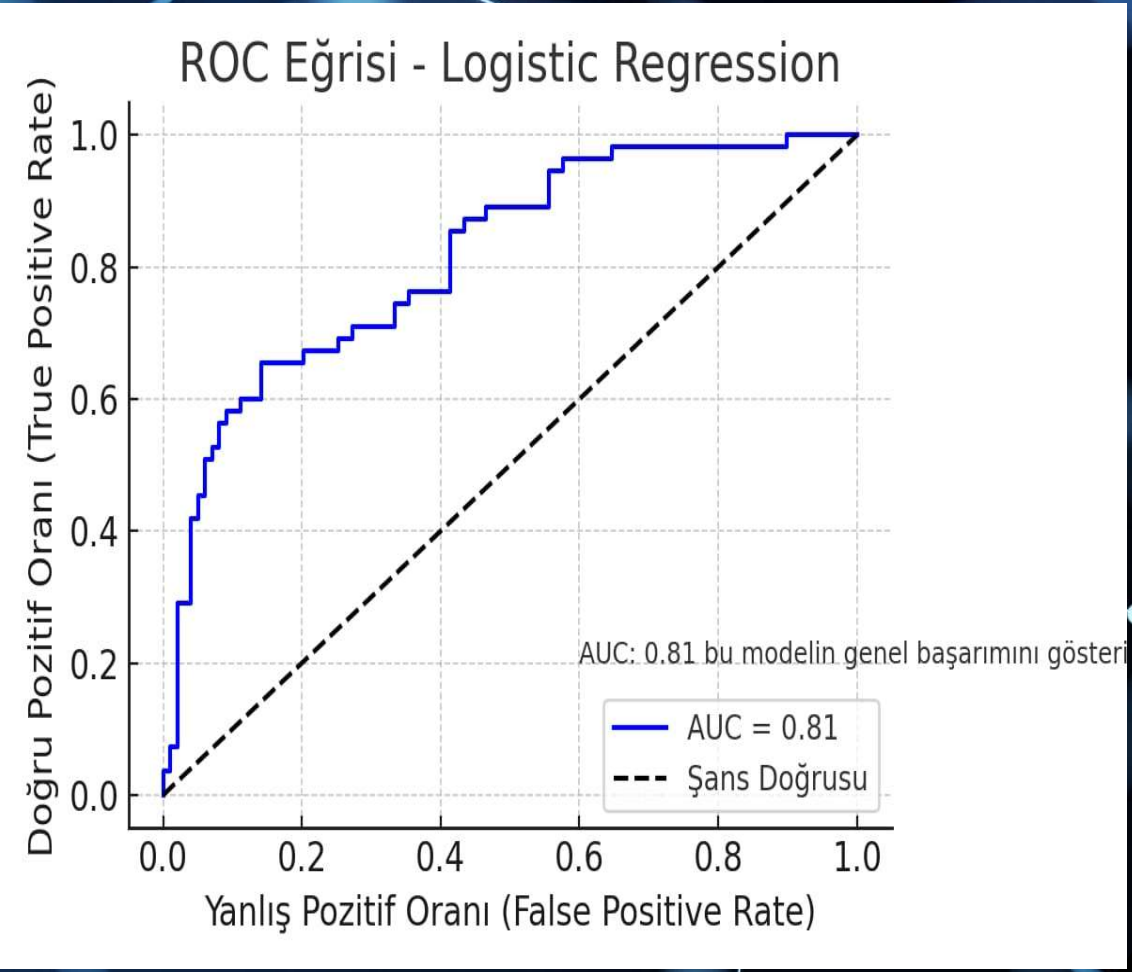
Model Evaluation – ROC Curve

Meaning of the Curve

The closer the curve is to the top-left corner, the better the model performs.

The AUC (Area Under the Curve) value summarizes the overall performance of the model as a single number.

AUC = 1.0 → Perfect model
AUC = 0.5 → Random guessing



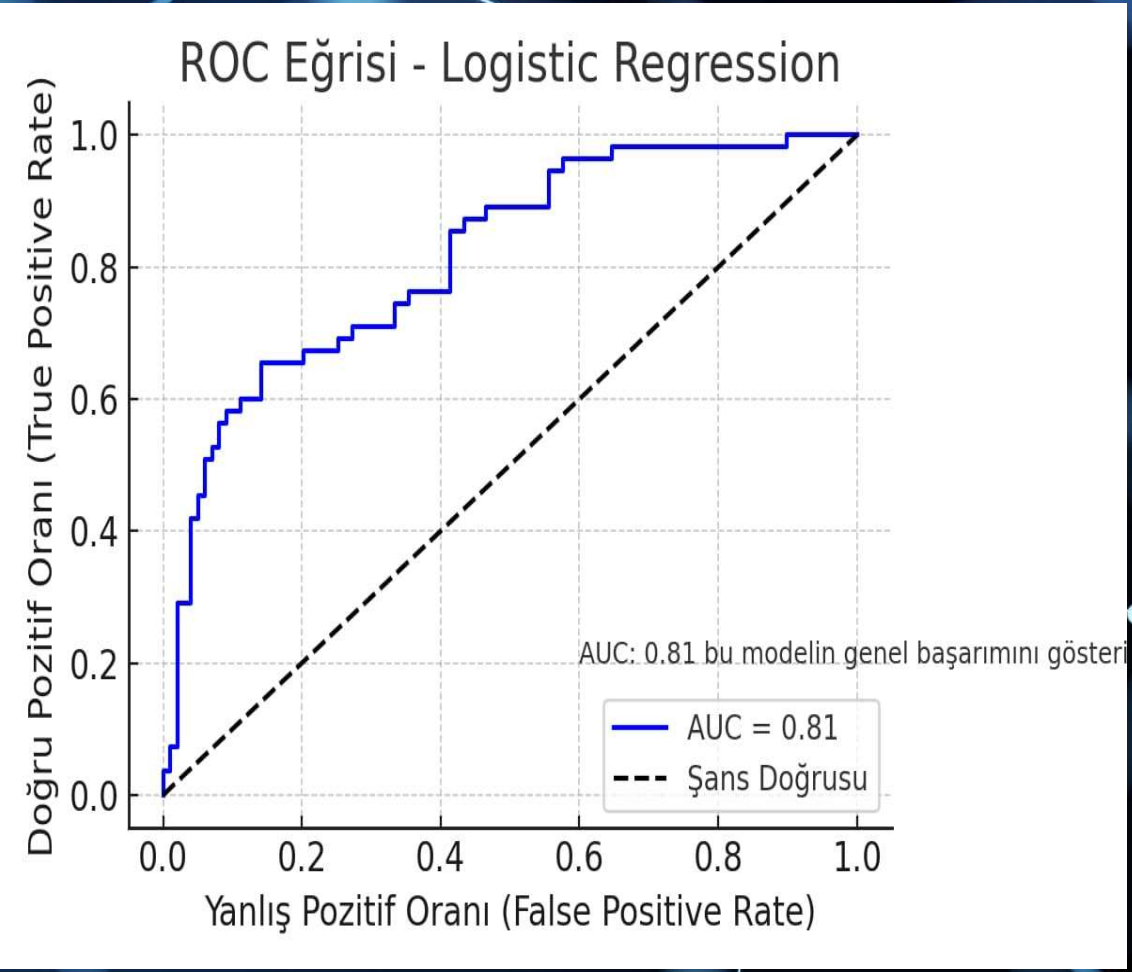
Model Evaluation – ROC Curve

Why Do We Look at This Chart?

To evaluate not only the “accuracy rate” but also the overall prediction balance.

To assess both sensitivity (recall) and false alarm rate together.

To understand the true performance of the model, especially in datasets with class imbalance.



Conclusion and Evalution

In this project, a machine learning system was developed to predict diabetes based on health indicators such as glucose level, BMI, blood pressure, and age.

Using the **Pima Indian Diabetes Dataset**, steps such as data preprocessing, correlation analysis, and modeling were performed.

Logistic Regression and **Support Vector Machine (SVM)** models were compared, and both demonstrated successful performance with **AUC ≈ 0.77**.

However:

Logistic Regression offers strong **interpretability**, especially important in healthcare where understanding feature impact is crucial.

SVM can produce sharper decision boundaries but is **less interpretable** and more resource-intensive for large datasets.

Given the linear nature of the data and the importance of interpretability in this project, **Logistic Regression** was deemed the more appropriate model.

Key Takeaway:

With well-prepared data and proper model selection, machine learning can serve as an effective and reliable tool for early diagnosis and decision support in healthcare.