Data Scaling

ENSF 444

Learn about the different ways to scale data

Discuss the impact of scaling data on supervised learning

These are the topics that will be covered in the nest lecture slides

For other non-linear supervised learning methods, we need to scale the data properly.

Which Non-Linear methods need scaling?

1. SVM (Support Vector machines)
2. K-nearest neighbours
3. Neural Networks
4. Regularized Methods

What are Regularized methods ?
From GPT

Regularized methods in machine learning are techniques used to prevent overfitting and improve the generalization ability of a model. Overfitting occurs when a model learns the training data too well, capturing noise or irrelevant patterns, which leads to poor performance on unseen data.

Regularized methods include L1(Lasso) and L2(Ridge) and then Elastic Net which is a combo.

Here are the main Preprocessing methods

StandardScaler

Mean of zero, variance of one

RobustScaler

Median of zero, interquartile(median) range of one

MinMaxScaler

Minimum of zero, maximum one

Normalize

Each sample (row) has unit norm

Standard Scalar

A technique for transforming numerical data to have a mean of zero and a standard deviation of one. Useful for Machine learning algos that perform better when input variables are scaled to a standard range. The formula for standard scaling is

$$z = \frac{x - \mu}{\sigma}$$

where
z is the scaled data
x is the original data
mew is the mean of the data
sigma is the standard deviation

What does this do ?
Transforms the data so that each feature the mean is 0 and the variance is 1

Kernel Density Estimation plot

This is a visual helper, almost like a histogram, like a continuous version of a histogram.

Boxplot

You learned this in 319 but its good to learn again.

Minimum Score - Smallest Value in dataset

Lower Quartile - Value below 25% of the data falls - known as the first quartile

Median - Middle value of the data set - Second Quartile

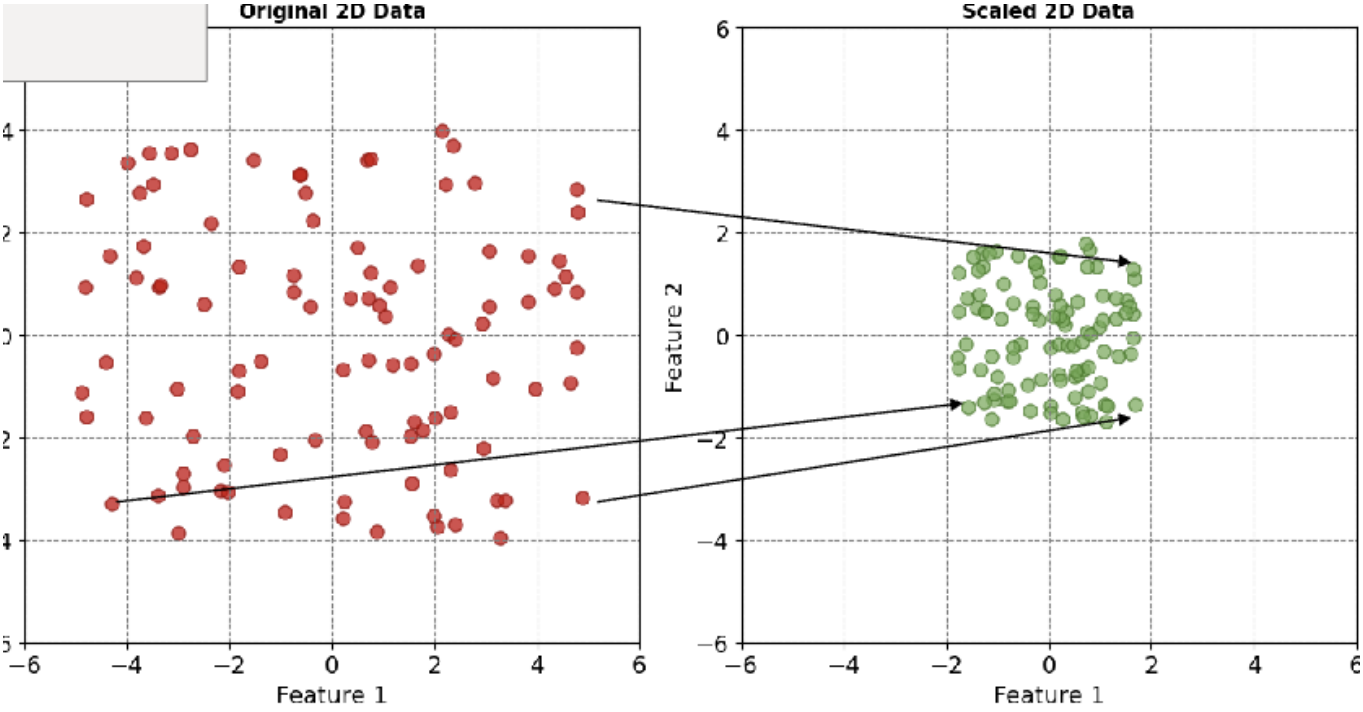
Upper Quartile - Value below which 75% of data falls - Third Quartile

Maximum Score - Largest value in data set

Whiskers - Lines extending

Interquartile - range between the first quartile and the third quartile

```
# Create 2D dataset
data_2d = np.random.rand(100, 2)
data_2d[:,0] = data_2d[:,0] * 10 - 5
data_2d[:,1] = data_2d[:,1] * 8 - 4
# Initialize the standard scaler
scaler = StandardScaler()
# Fit and transform the 2D dataset
scaled_data_2d =
scaler.fit_transform(data_2d)
```



This is what standard scaling does to the data

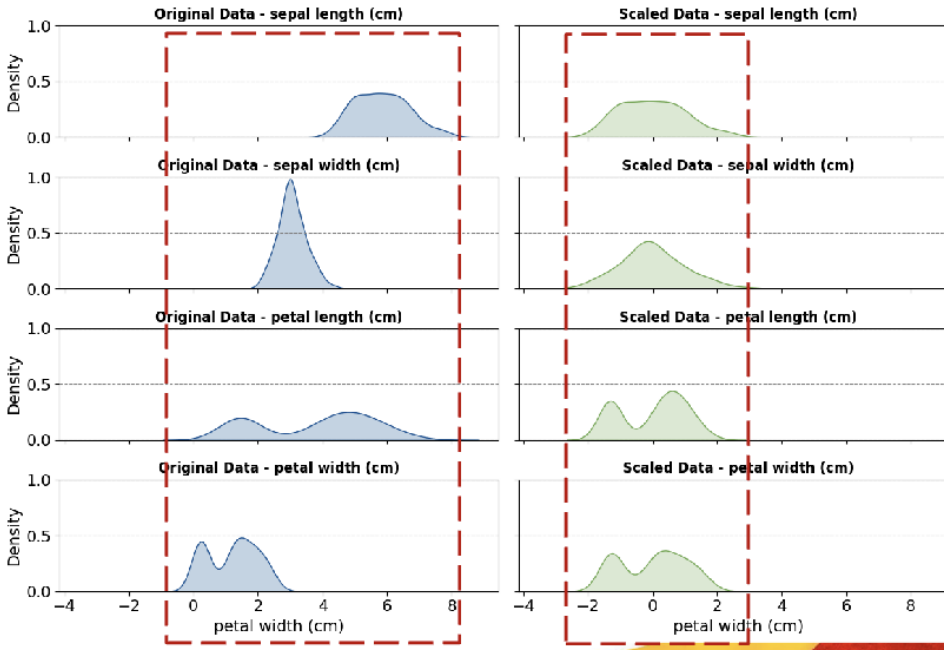
Heres an example of the iris data set being scaled

Iris Dataset

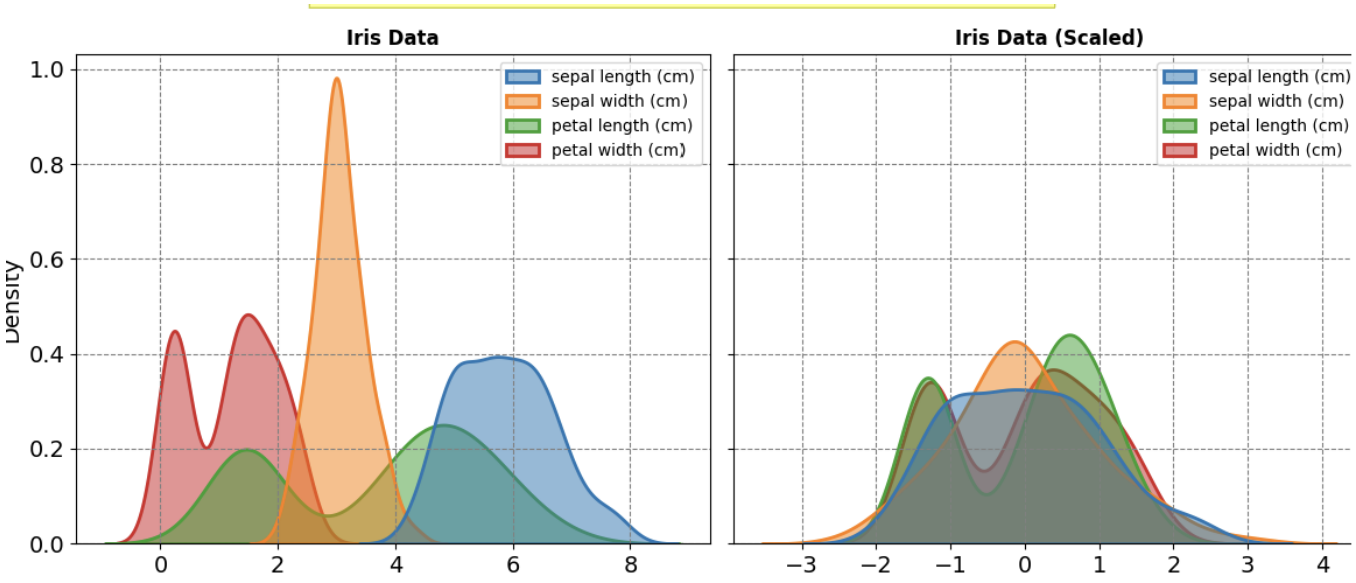
```
 Load Iris dataset
ris = load_iris()

 Create DataFrame for
isualization
ata =
d.DataFrame(data=iris.data,
olumns=iris.feature_names)

 StandardScaler
caler = StandardScaler()
caled_data =
caler.fit_transform(data)
```



it is most visible in the petal length data



Here is a comparison - of the data being scaled vs original data

---

Robust Scalar Method

Given a dataset, **for each feature**, This method adjusts the values using a formula

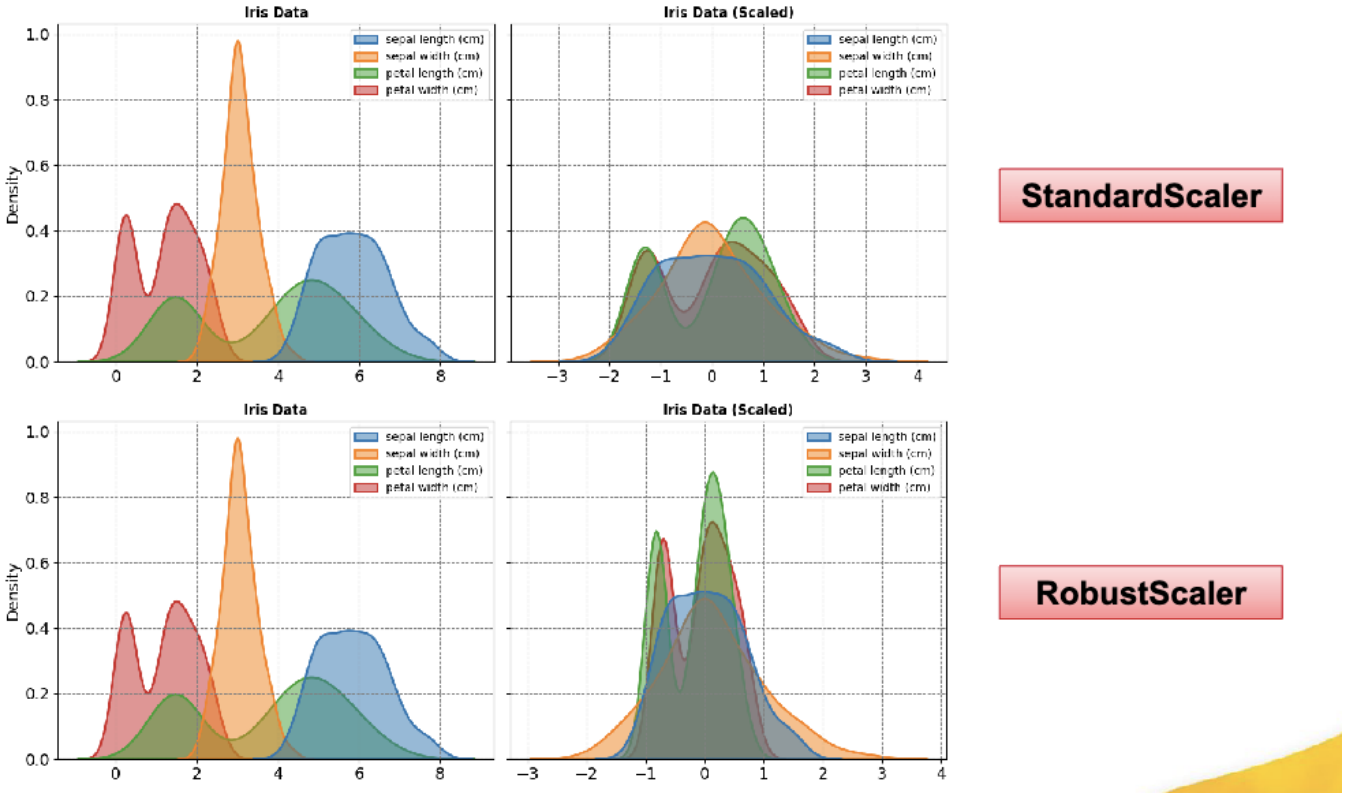
$$x_{scaled} = \frac{x - median(x)}{IQR(x)}$$

Where

x is the original feature vector
median(x) is the Median of the feature vector
IQR - interquartile range of the feature vector which is Q3 - Q2
x scaled is the scaled feature vector

This makes robust scaler less prone to outliers. This method is used when data has outliers. its a way to standardize the data that is robust to outliers.

The goal is to not remove outliers but to ensure that they have less influence on scaled data. Thus can be helpful for ml models that are sensitive to the range of the input features



Here is a comparison VS StandardScaler to Robust.

MinMax Scaler Method

This method is for data normalization.

Normalizing data refers to the process of scaling individual samples to have a mean of zero and a standard deviation of one. This is so that features in machine learning are ensured that they are similar to scale.

1. **Calculate Mean and Standard Deviation:** For each feature, calculate its mean and standard deviation across all samples in the dataset.
2. **Subtract Mean:** Subtract the mean value of the feature from each sample. This centers the data around zero.
3. **Divide by Standard Deviation:** Divide each centered sample by the standard deviation of the feature. This scales the data so that it has a standard deviation of one.

By normalizing the data, you make the features more comparable and prevent features with larger scales from dominating those with smaller scales during the training process. This can lead to faster convergence and more stable performance, especially for algorithms that are sensitive to the scale of input features, such as gradient descent-based optimization algorithms.

This method scales each feature to a given range, which is usually 0 - 1

Here is the formula used

$$x_{scaled} = \frac{x - min(x)}{max(x) - min(x)}$$

where

x is the original feature vector

min(x) is the minimum value of the feature vector
max(x) is the maximum value of the feature vector
x scaled is the scaled feature vector

This scaling method is beneficial when you want your data to be bounded within a certain range. But an important note is that MInMaxScaler does not reduce the impact of outliers. So if we are dealing with data that has outliers, use RobustScaler.

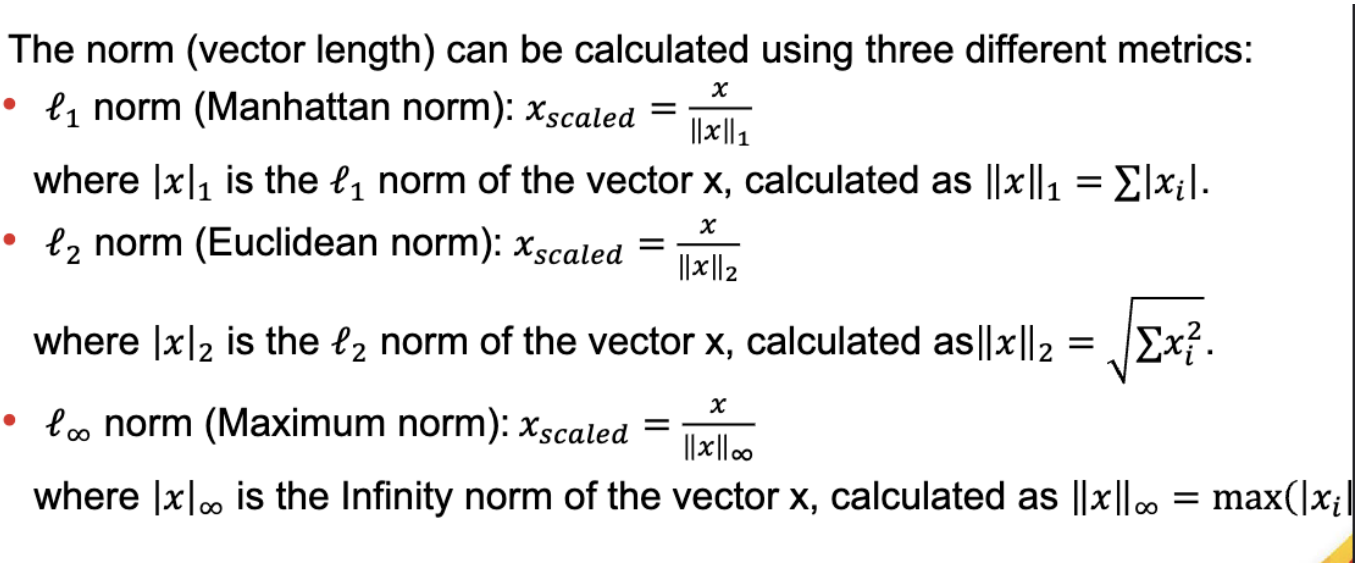
Normalizer Method

Different kind of rescaling

Each feature or data sample with at least one non-zero component is rescaled independently of other features or samples so that its norm (vector length) equals one.

The current version of Normalizer scales each **row** of the data to unit form. this is the version i tried to use first in the final project and then changed to standard scaler

This can be particularly useful in certain applications like text classifications or clustering where cosine is similarity is used.



Here is the formula summary

Manhattan norm
Euclidean norm
Maximum norm

How to implement Scaling

5 Steps

1. Import scaling method
2. Instantiate Scaler
3. Fit the scaler to the training data (only the feature matrix)
4. Transform the training data (only the feature matrix)
5. Transform the testing data (only the feature matrix)

Implementing Scaling
To implement proper scaling, fit scaler on the training data, then transform both the training data and the testing data
DO NOT fit the scaler with the testing data, data will be changed and will not have the same relationship as the training data