

The background features a light purple and white color scheme with large, soft-edged abstract shapes. Four detailed illustrations of purple iris flowers are placed around the central text: one in the top-left, one in the top-right, one in the bottom-left, and one in the bottom-right. Each flower has yellow centers and green leaves. Two clusters of small, dark blue dots are also present, one on the left and one on the right side of the page.

Iris Flower Classification

Completed By: Sofia Rueda and Denise Campos
DATA 3421
Spring '25



Intro


The Iris dataset is one of the most well-known and widely used datasets in machine learning. It contains measurements of three different species of Iris flowers; Setosa, Versicolor, and Virginica, and is commonly used to demonstrate classification techniques.

Can we accurately classify flower species based on their physical measurements using basic machine learning algorithms?

Can we identify iris species by features like sepal and petal dimensions?

Can we determine characteristic value ranges for each species?

These questions motivated us to conduct exploratory data analysis (EDA) and build predictive models. The results can aid in species identification and, if refined further, our models could be adapted for classification problems in other fields as well.



What does our data look like?



	sepal length (cm)	sepal width (cm)	petal length (cm)	petal width (cm)	target	target_name
0	5.1	3.5	1.4	0.2	0	setosa
1	4.9	3.0	1.4	0.2	0	setosa
2	4.7	3.2	1.3	0.2	0	setosa
3	4.6	3.1	1.5	0.2	0	setosa
4	5.0	3.6	1.4	0.2	0	setosa

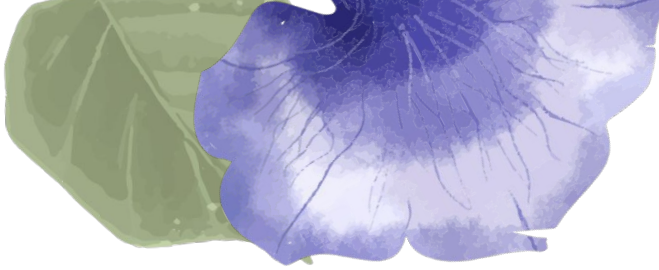
4 Attributes: sepal length, sepal width, petal length, petal width.

1 target: target_name (setosa, versicolor, virginica)

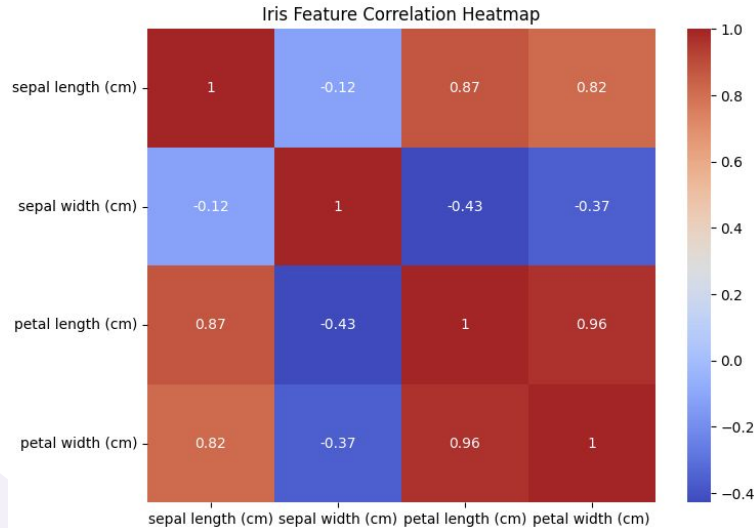
The Attributes data type is float.

The Target's data type is object.

There is no class imbalance.



Data Specs



The petal features seem to be highly correlated.

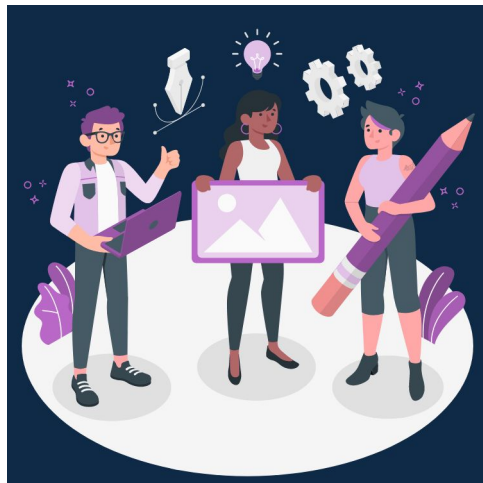
We had 3 positively related features, and 3 inversely related features.

The most related features was petal width and petal length with a correlation of .96



Data cleaning

- 1 duplicate row, removed with `drop_duplicates()`
- Disregarding “target” column

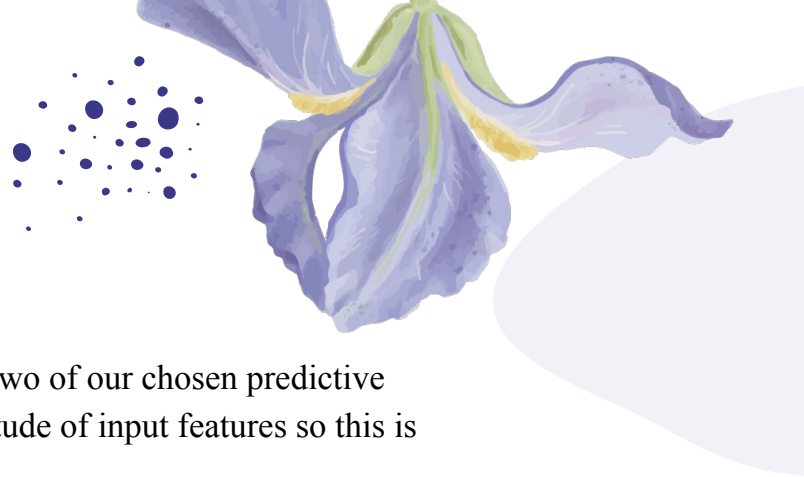




The models we used are:

- **KNN** ➡ A simple model that works well when classes are well separated.
- **Logistic Regression** ➡ A linear model that is great for multiclass classification. It gives us easy to interpret outputs.
- **Decision Tree** ➡ A rule based model that can handle nonlinear relationships, and is easy to visualize.

Preprocessing



At this point in our analysis, we standardized the feature values. Two of our chosen predictive models, [KNN](#) and [Logistic Regression](#), are sensitive to the magnitude of input features so this is a necessary step.

We split the dataset into 80% training and 20% testing.

A single train-test split can be biased, so we used 5-fold cross-validation to get a more reliable evaluation. The data was split into 5 parts, and each model was trained and tested 5 times, rotating the test set each time. We then averaged the results for a more robust performance estimate.

With this method, we achieved average accuracies of 97% for KNN and Logistic Regression, and 95% for the Decision Tree. These results gave us confidence that our models weren't just memorizing the training set, but they were generalizing well across different subsets of the data.

Why Cross Validation?

- In our case, we are working with an extremely small data, with only 150 samples.
- Its small size makes it really easy to overfit.
- 5-fold cross-validation helped us avoid that by making sure every sample had a chance to be in the test set. This gave us a more reliable sense of how well each model would perform on truly unseen data.



We added Noise

This was to simulate real world imperfections. Our data came pretty much clean and that is not the case for real world data analysis.

This step helped us evaluate how robust each model was.

A good model will perform well even when data is NOT perfect.

The results:

- Slight drop in accuracy for KNN (83%) and Logistic regression (86%)
- Decision tree accuracy stayed at 100%



◆ KNN Classification Report:

	precision	recall	f1-score	support
setosa	1.00	1.00	1.00	10
versicolor	0.70	0.78	0.74	9
virginica	0.80	0.73	0.76	11
accuracy			0.83	30
macro avg	0.83	0.84	0.83	30
weighted avg	0.84	0.83	0.83	30

◆ Logistic Regression Classification Report:

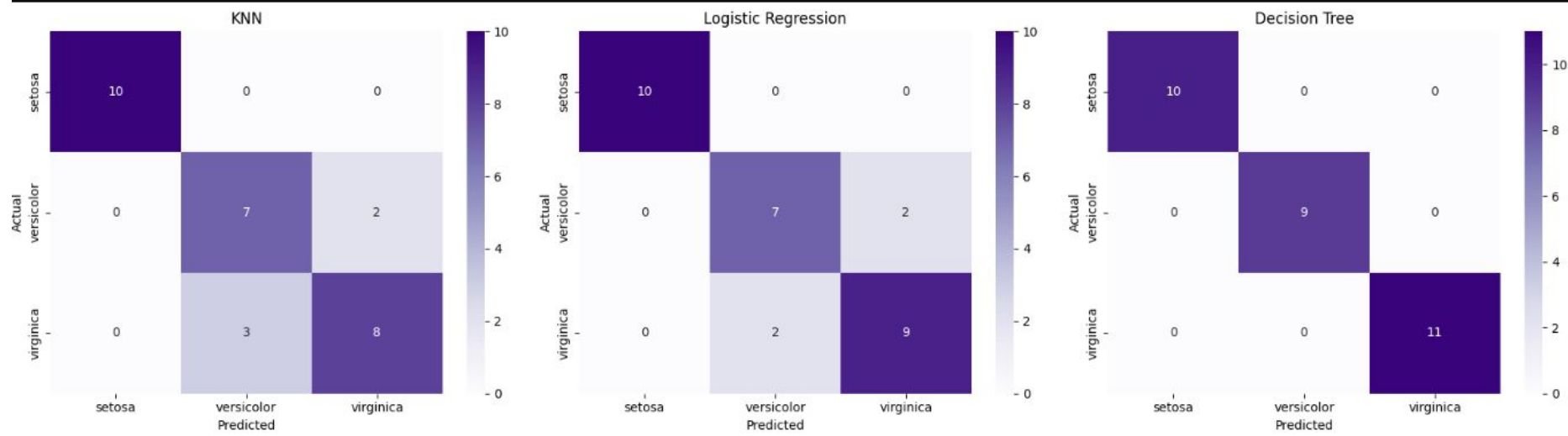
	precision	recall	f1-score	support
setosa	1.00	1.00	1.00	10
versicolor	0.78	0.78	0.78	9
virginica	0.82	0.82	0.82	11
accuracy			0.87	30
macro avg	0.87	0.87	0.87	30
weighted avg	0.87	0.87	0.87	30

◆ Decision Tree Classification Report:

	precision	recall	f1-score	support
setosa	1.00	1.00	1.00	10
versicolor	1.00	1.00	1.00	9
virginica	1.00	1.00	1.00	11
accuracy			1.00	30
macro avg	1.00	1.00	1.00	30
weighted avg	1.00	1.00	1.00	30

There is clear overfitting with the Decision Tree model. 📢

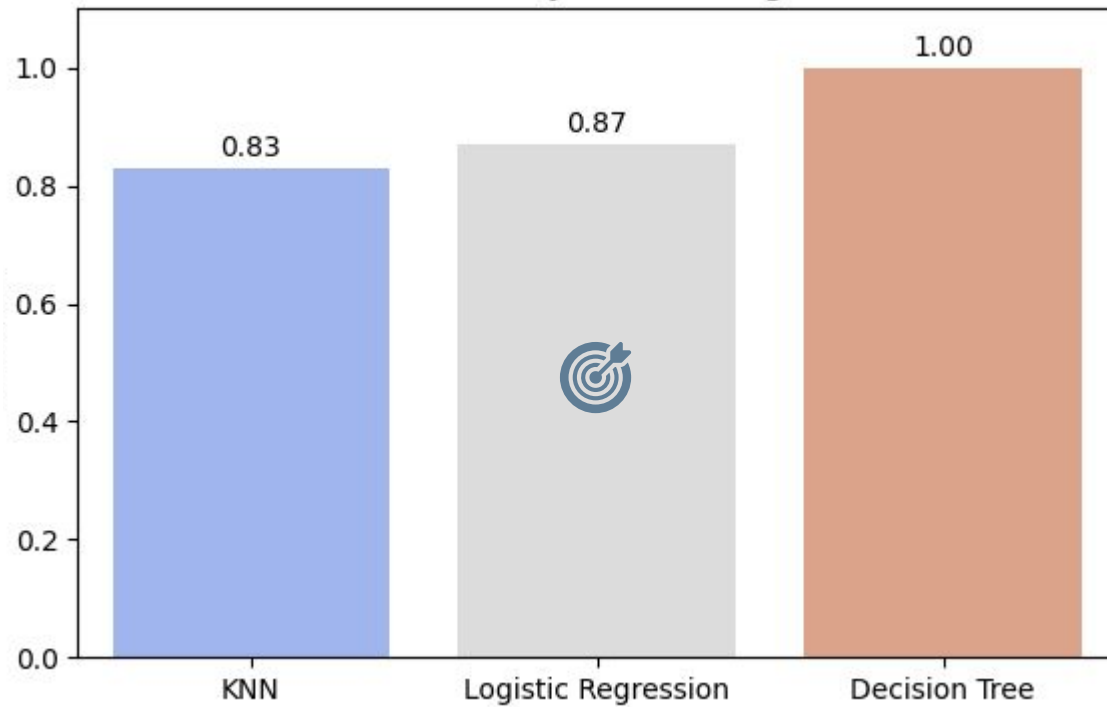
The higher overall accuracy and f1-scores for Logistic Regression makes it the best model. 📈






Each confusion matrix shows how many flowers were correctly or incorrectly classified for each species. 🔍?

Each model perfectly predicted Setosa, but struggled with Versicolor and Virginica. This can suggest that these 2 species are harder to identify and models may need further tuning to discover meaningful differences. 🌸

Model Accuracy After Adding Noise





Since Logistic Regression was our best model, we wanted to find which feature held the most prediction power.

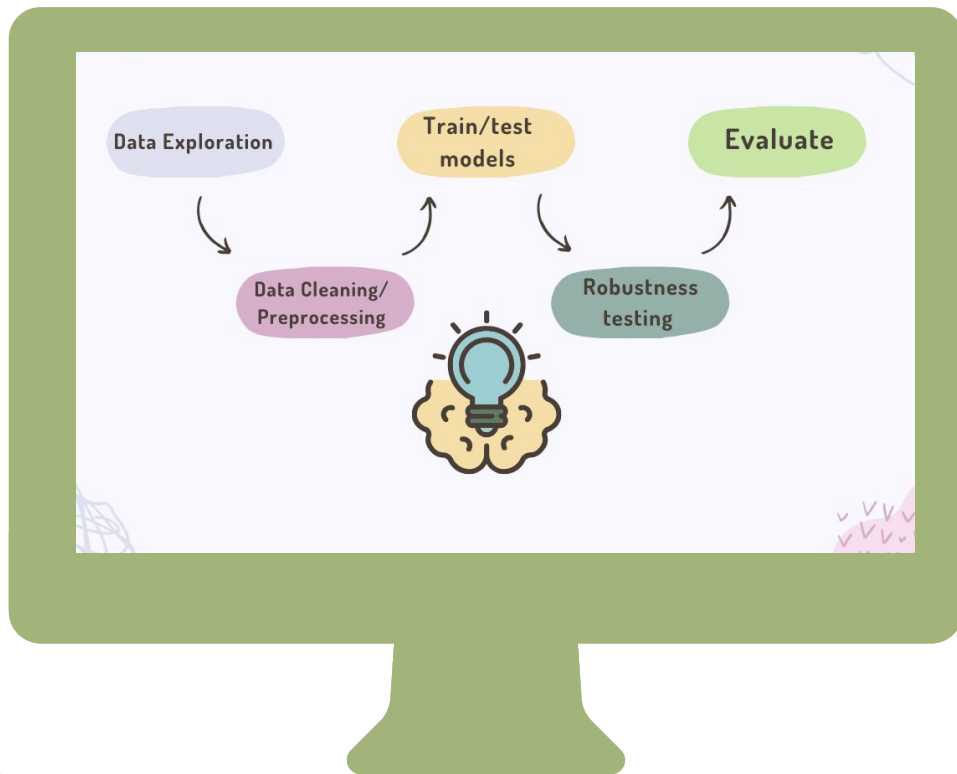
Even with this discovery, it **does not** imply causation. Further study is needed to deduce that.

The feature with the largest correlation coefficient was petal width.

	Feature	Importance
3	petal width (cm)	1.608433
2	petal length (cm)	1.434637
1	sepal width (cm)	0.763249
0	sepal length (cm)	0.668777

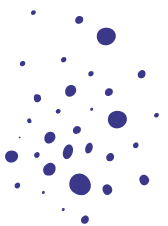


Pipeline

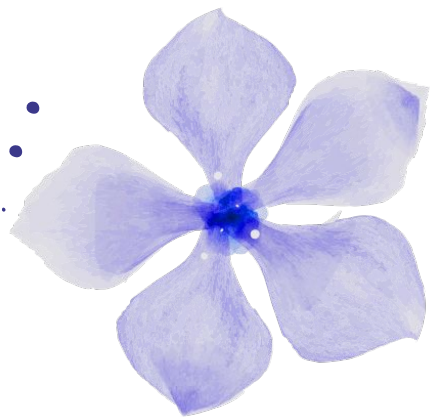


Key Points

- Data's target was balanced
- Data's features were highly correlated
- We chose KNN, Logistic Regression, and Decision Tree models to try
- Conducted a cross validation due to small dataset size
- Added noise to test model robustness
- Decision tree model was overfitting
- KNN and Logistic Tree models both had good accuracies and f1-scores
- The best model, in terms of highest accuracy and f1 score, was the Logistic Regression model
- The feature with the largest prediction power was petal width.



Thank you for listening.



REFERENCE

Fisher, R. A. (1936). The use of multiple measurements in taxonomic problems. *Annals of Eugenics*, 7(2), 179-188.
Iris dataset retrieved from the UCI Machine Learning Repository: <https://archive.ics.uci.edu/ml/datasets/iris>.