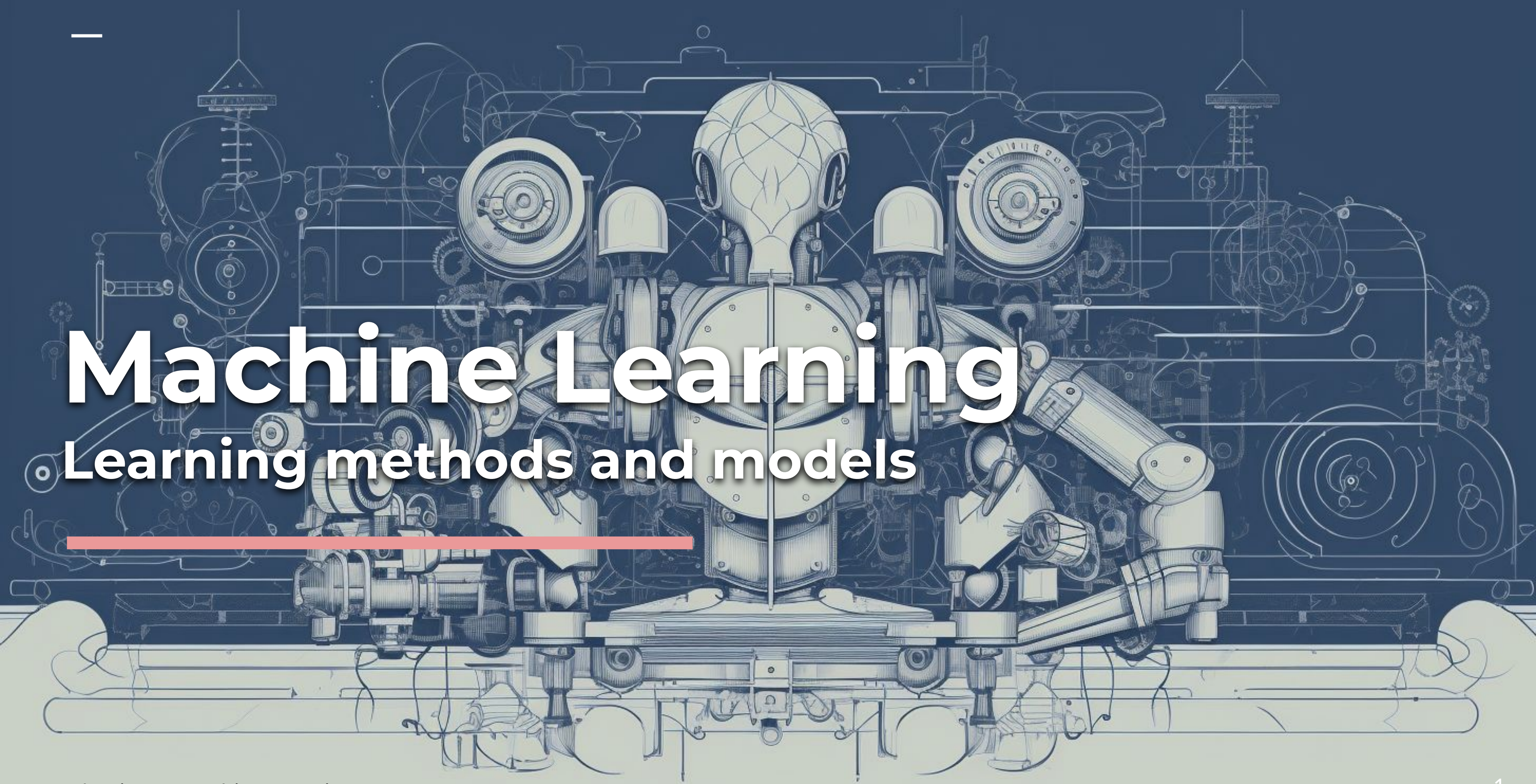


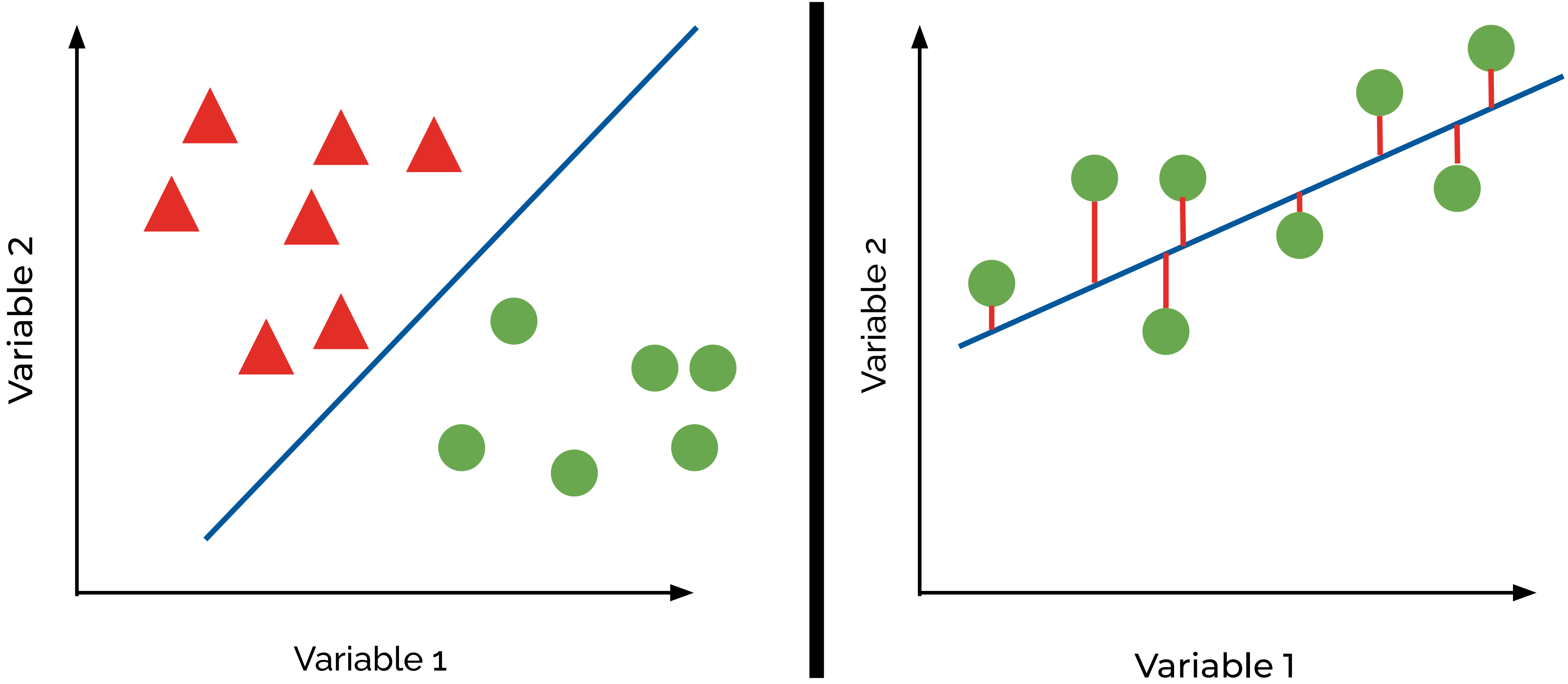
Machine Learning

Learning methods and models

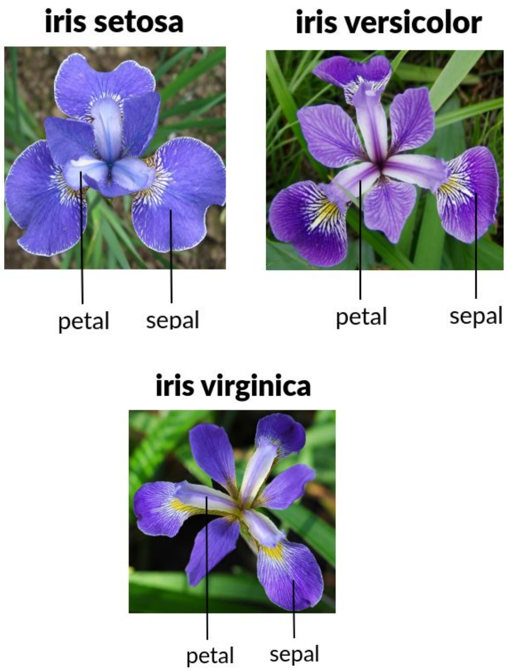
Tasks

Classification and regression

Classification and regression



Example of classification

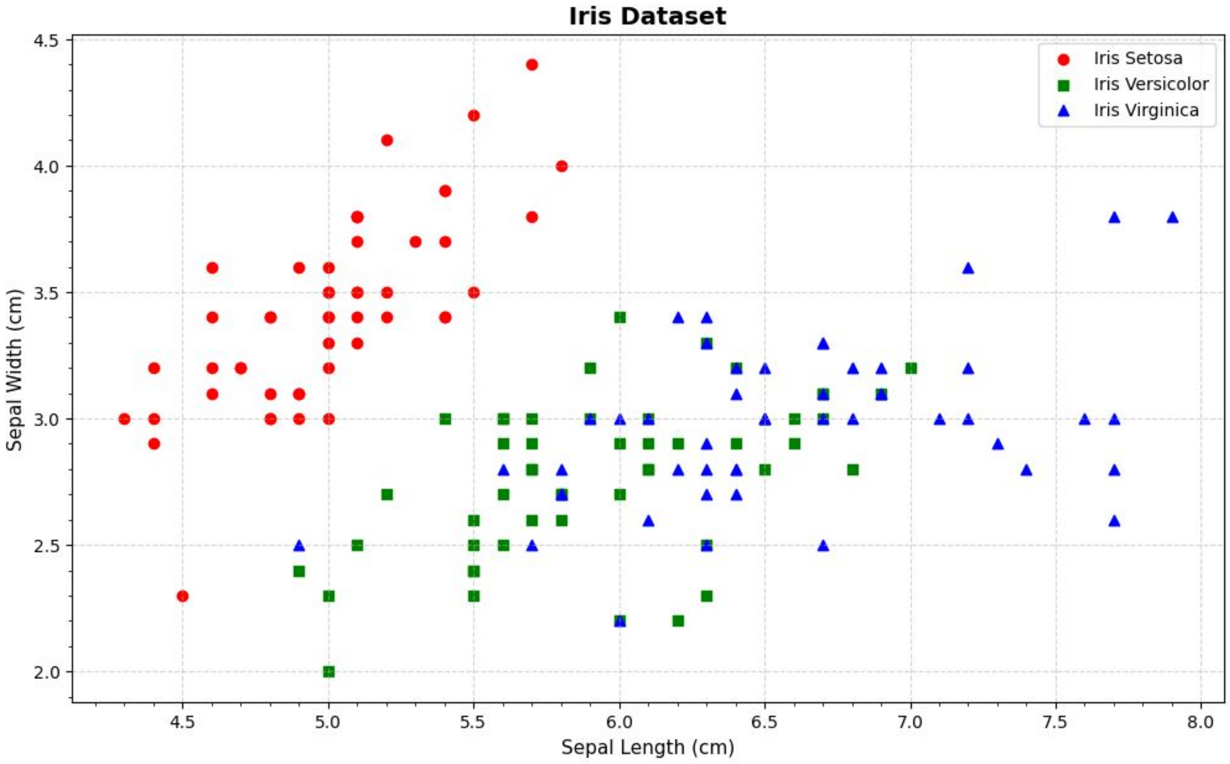


- Three different species.
- Described with the characteristics of its sepals and petals.
    - Sepal width
    - Sepal length
    - Petals width
    - Petals length
- 150 examples, 50 per species.
- Task: find a model that predicts the specy given the four characteristics.

Example of classification

```
from sklearn import datasets


# Load the Iris dataset
iris = datasets.load_iris()
X = iris.data[:, :2]
y = iris.target
```

Example of classification



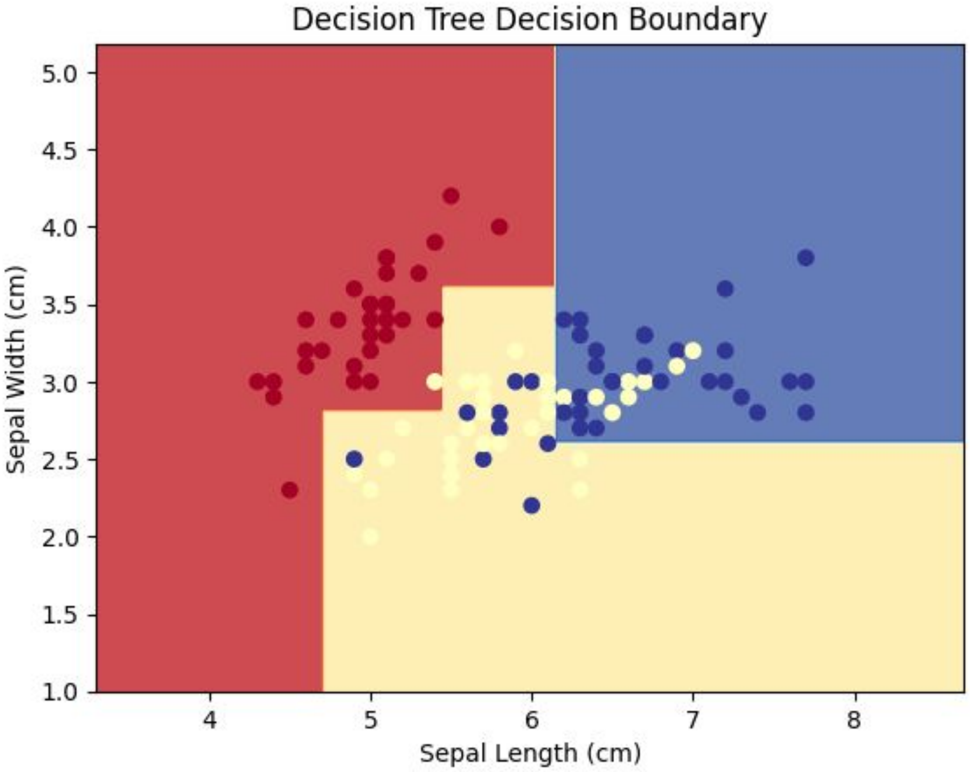
Solving the IRIS classification task

```
# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X, y,
test_size=0.3, random_state=42)

# Standardize the features (optional but recommended)
scaler = StandardScaler()
X_train = scaler.fit_transform(X_train)
X_test = scaler.transform(X_test)
```

Solving the IRIS classification task

```
# Train and visualize a Decision Tree classifier
dtree = DecisionTreeClassifier(max_depth=3)
dtree.fit(X_train, y_train)
accuracy_dtree = dtree.score(X_test, y_test)
print(f"Decision Tree Accuracy: {accuracy_dtree:.2f}")


# Decision Tree Accuracy: 0.76
```

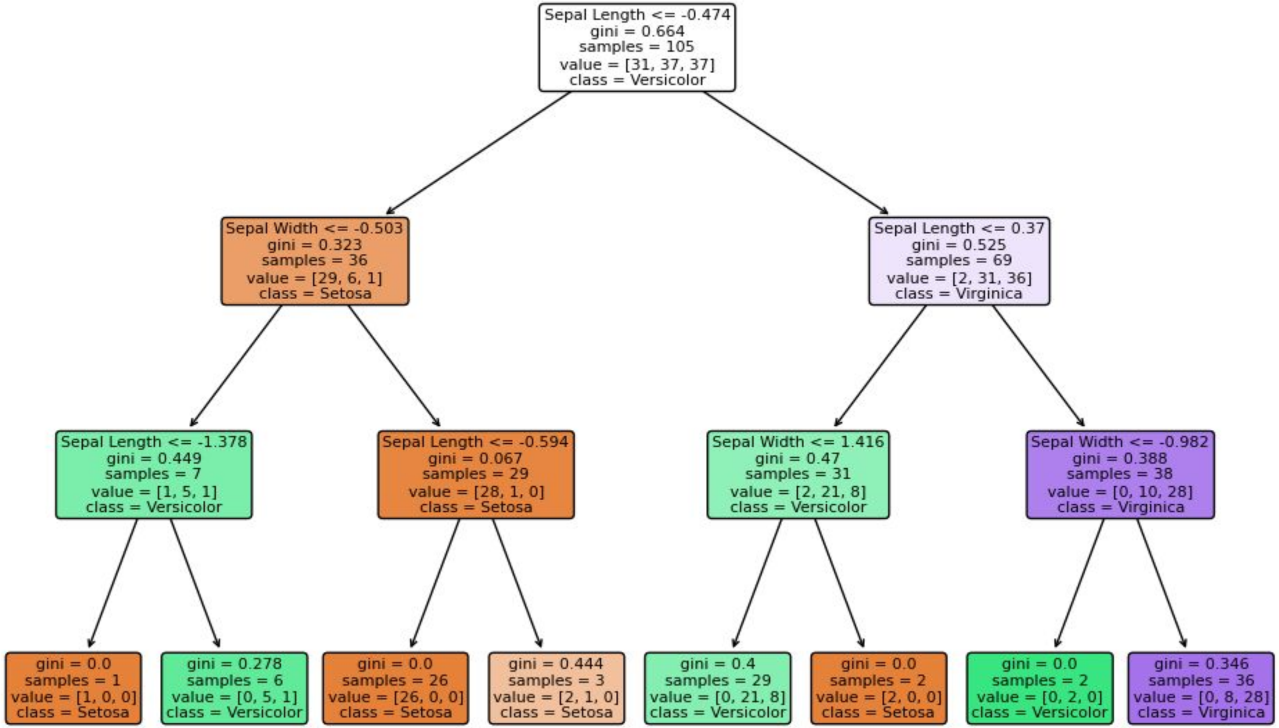
Solving the IRIS classification task



Solving the IRIS classification task

```
from sklearn.tree import plot_tree


# Visualize the decision tree
plt.figure(figsize=(12, 8))  # Set the figure size
plot_tree(dtree, filled=True, feature_names=['Sepal Length', 'Sepal
Width'], class_names=['Setosa', 'Versicolor', 'Virginica'],
rounded=True, fontsize=8)
plt.title('Decision Tree Visualization')
plt.show()
```

Solving the IRIS classification task



Solving the IRIS classification task

```
from sklearn.tree import export_text

# Extract and display the rules
tree_rules = export_text(dtree,
feature_names=iris.feature_names[:2])
print("Decision Tree Rules:\n", tree_rules)
```

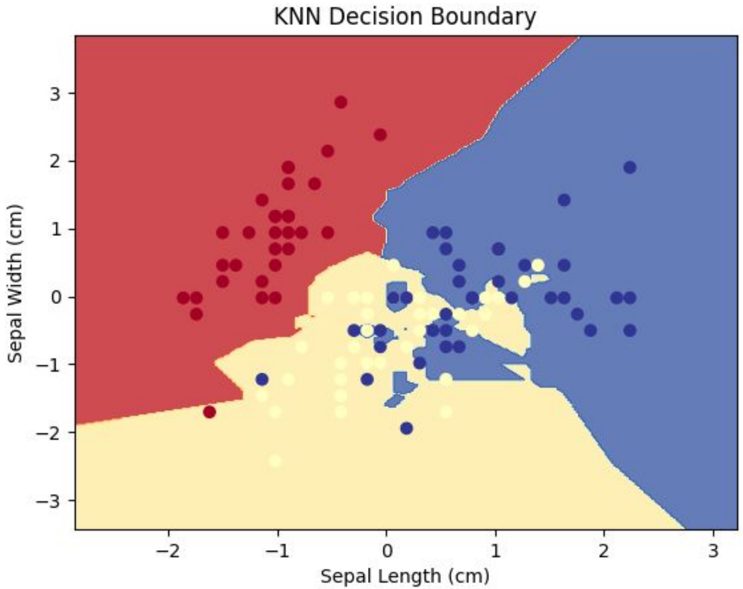
Solving the IRIS classification task

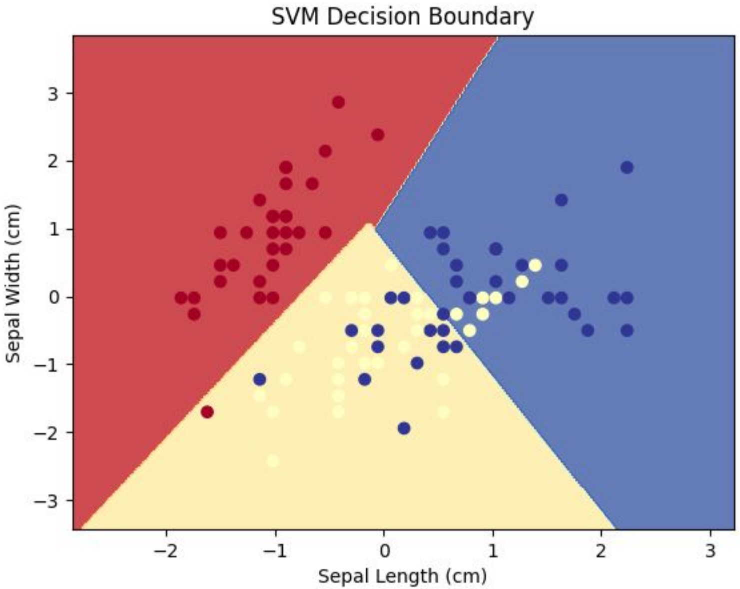
```
Decision Tree Rules:
 |--- sepal length (cm) <= 5.45
|   |--- sepal width (cm) <= 2.80
|   |   |--- sepal length (cm) <= 4.70
|   |   |   |--- class: 0
|   |   |--- sepal length (cm) >  4.70
|   |   |   |--- class: 1
|   |--- sepal width (cm) >  2.80
|   |   |--- sepal length (cm) <= 5.35
|   |   |   |--- class: 0
|   |   |--- sepal length (cm) >  5.35
|   |   |   |--- class: 0
|--- sepal length (cm) >  5.45
|   |--- sepal length (cm) <= 6.15
|   |   |--- sepal width (cm) <= 3.60
|   |   |   |--- class: 1
|   |   |--- sepal width (cm) >  3.60
|   |   |   |--- class: 0
|   |--- sepal length (cm) >  6.15
|   |   |--- sepal width (cm) <= 2.60
|   |   |   |--- class: 1
|   |   |--- sepal width (cm) >  2.60
|   |   |   |--- class: 2
```

Solving the IRIS classification task

```
# Define new samples with feature values
new_samples = np.array([[5.1, 3.5],  # Sample 1
                       [6.2, 2.9],  # Sample 2
                       [7.3, 2.8]])  # Sample 3

# Predict the class labels for the new samples
predicted_classes = dtree.predict(new_samples)

print("predicted_classes")
# array([0, 2, 2])
```

Solving the IRIS classification task

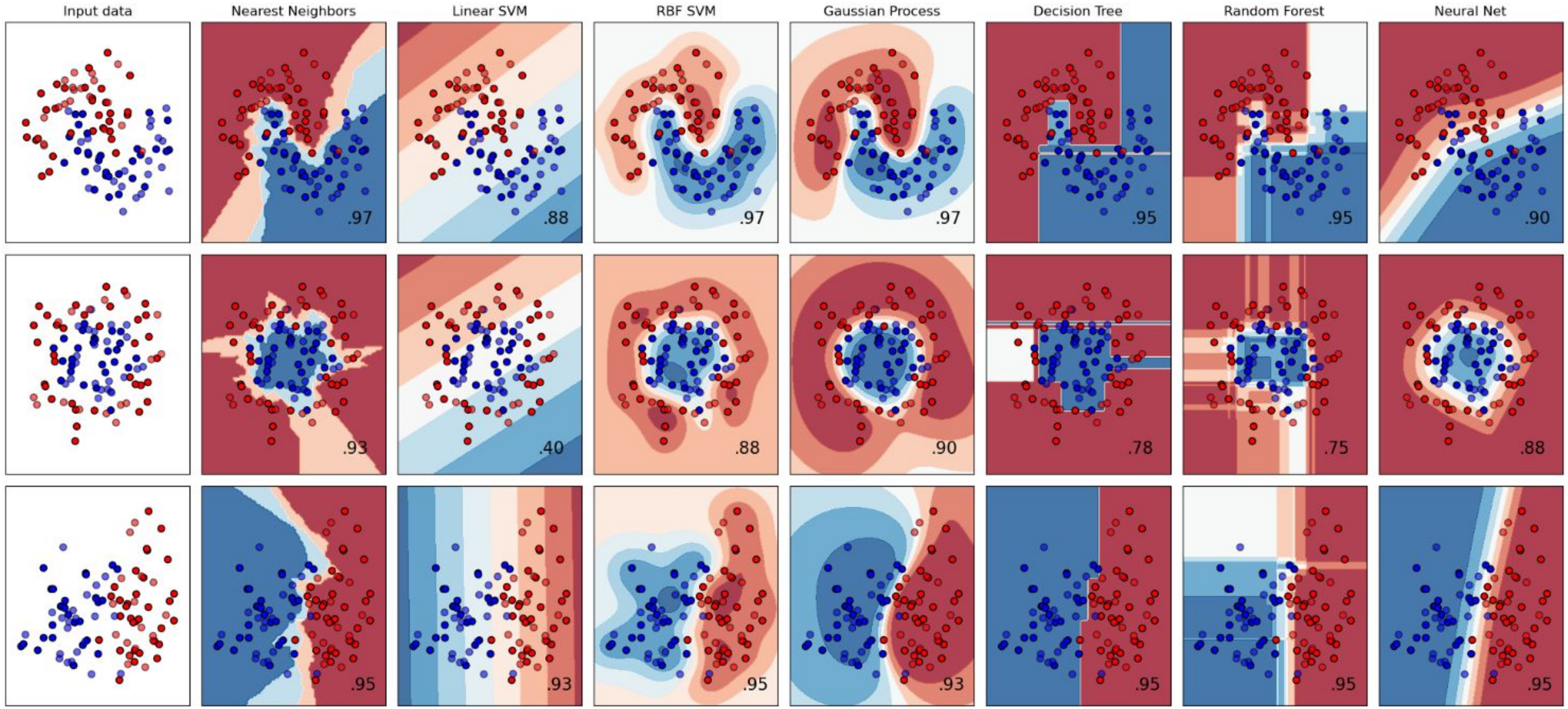




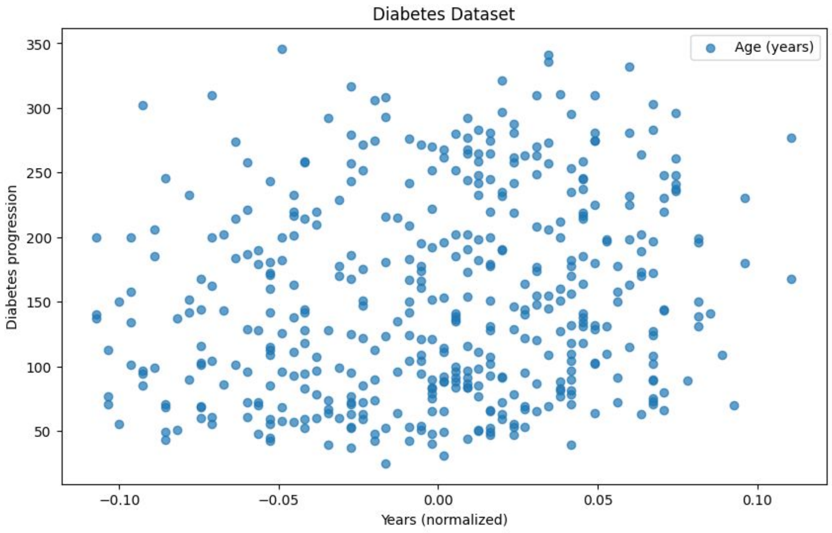
Solving the IRIS classification task

https://colab.research.google.com/drive/1Kx0okD6FHC5O74bblgC96h47DPH6Nson?usp=sharing

Solving the IRIS classification task

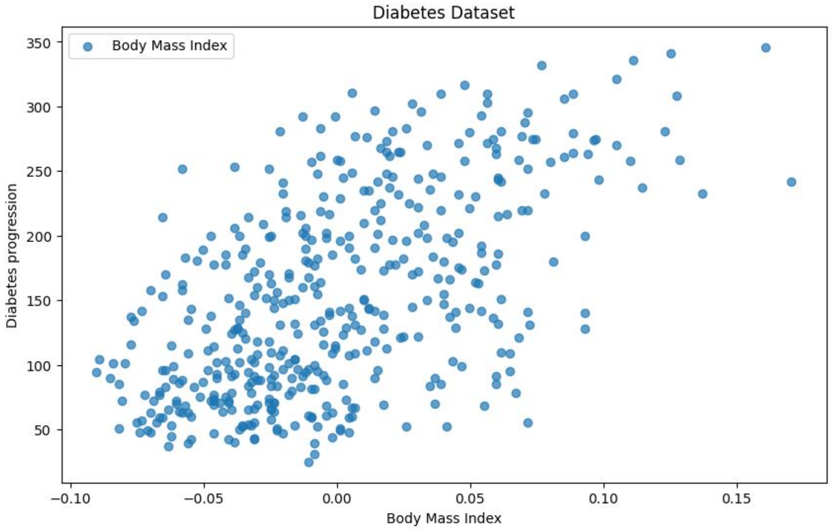


Example of regression



- Diabetes progression
- 10 features.
  - age age in years
  - sex
  - bmi body mass index
  - bp average blood pressure
  - s1 tc, total serum cholesterol
  - s2 ldl, low-density lipoproteins
  - s3 hdl, high-density lipoproteins
  - s4 tch, total cholesterol / HDL
  - s5 ltg, possibly log of serum triglycerides level
  - s6 glu, blood sugar level
- 442 examples
- Task: find a model that predicts diabetes progression.

Example of regression



- Diabetes progression
- 10 features.
  - age age in years
  - sex
  - bmi body mass index
  - bp average blood pressure
  - s1 tc, total serum cholesterol
  - s2 ldl, low-density lipoproteins
  - s3 hdl, high-density lipoproteins
  - s4 tch, total cholesterol / HDL
  - s5 ltg, possibly log of serum triglycerides level
  - s6 glu, blood sugar level
- 442 examples
- Task: find a model that predicts diabetes progression.

Example of regression

```
from sklearn.datasets import load_diabetes


# Load the diabetes dataset
diabetes = load_diabetes()
X = diabetes['data']
y = diabetes['target']
```

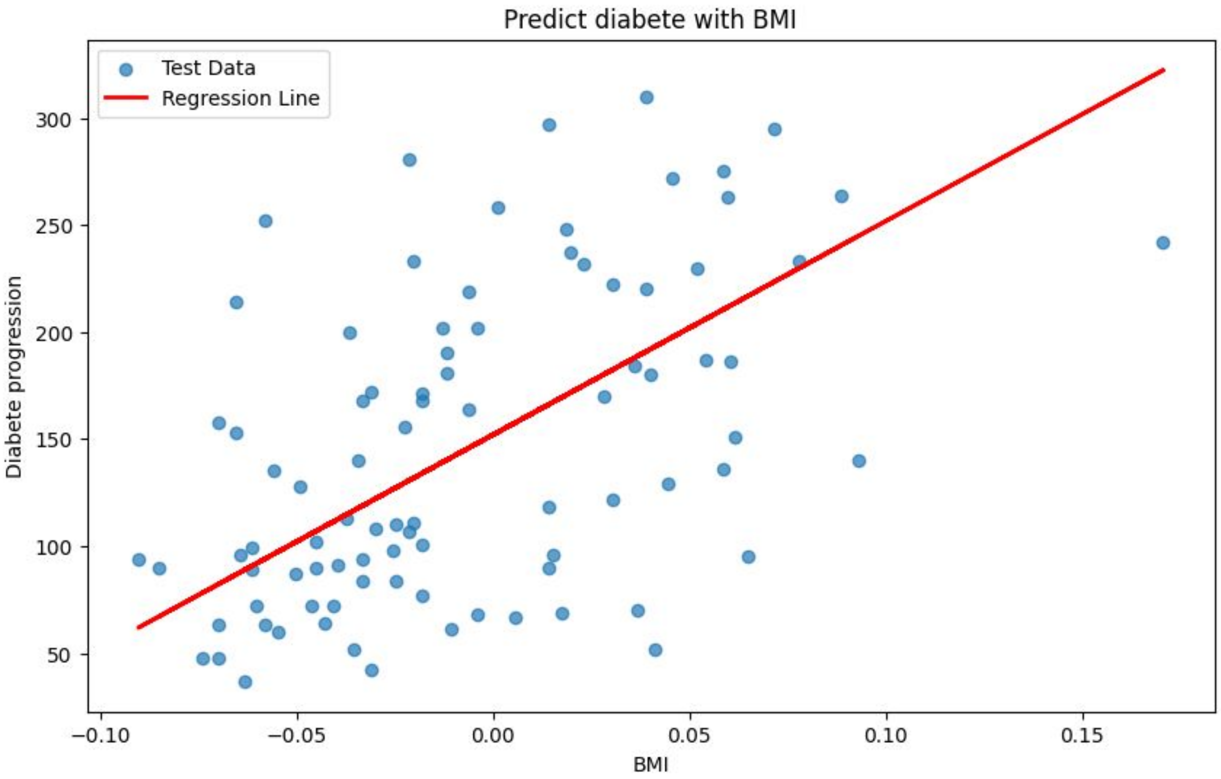
Example of regression

```python
# Choose a feature for the regression (e.g., feature 2)
X_selected = X[:, 2]

# Split the dataset into training and testing sets
X_train, X_test, y_train, y_test = train_test_split(X_selected, y,
test_size=0.2, random_state=42)

# Train a linear regression model
regression_model = LinearRegression()
regression_model.fit(X_train.reshape(-1, 1), y_train)
```

Example of regression

```
# Make predictions on the test set
y_pred = regression_model.predict(X_test.reshape(-1, 1))

# Evaluate the regression model
mse = mean_squared_error(y_test, y_pred)
r2 = r2_score(y_test, y_pred)
print(f"Mean Squared Error (MSE): {mse:.2f}")
print(f"R-squared (R2): {r2:.2f}")
# Mean Squared Error (MSE): 4061.83
# R-squared (R2): 0.23
```

Example of regression



Example of regression

```python
# Make a prediction for a new data point
new_data_point = np.array([0.05])  # Adjust the feature value as needed
predicted_value = regression_model.predict(new_data_point.reshape(-1, 1))
print(f"Predicted Value for New Data Point: {predicted_value[0]:.2f}")

# Predicted Value for New Data Point: 201.93
```

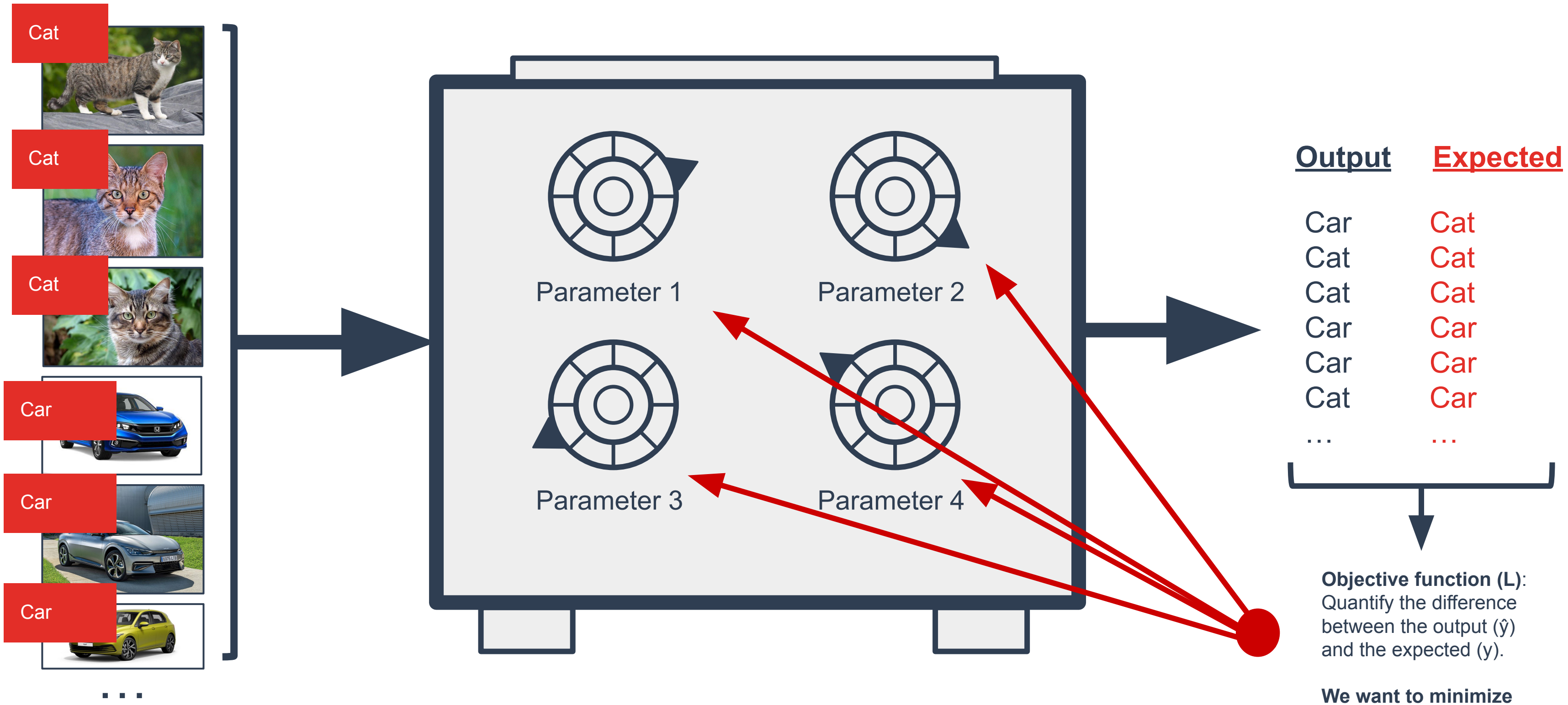
Solving the IRIS classification task

https://colab.research.google.com/drive/1i2To3ijnBU6uDnOqefWxsUjips0DWbF3?usp=sharing

Models

Parametric and non-parametric models

Parametric models

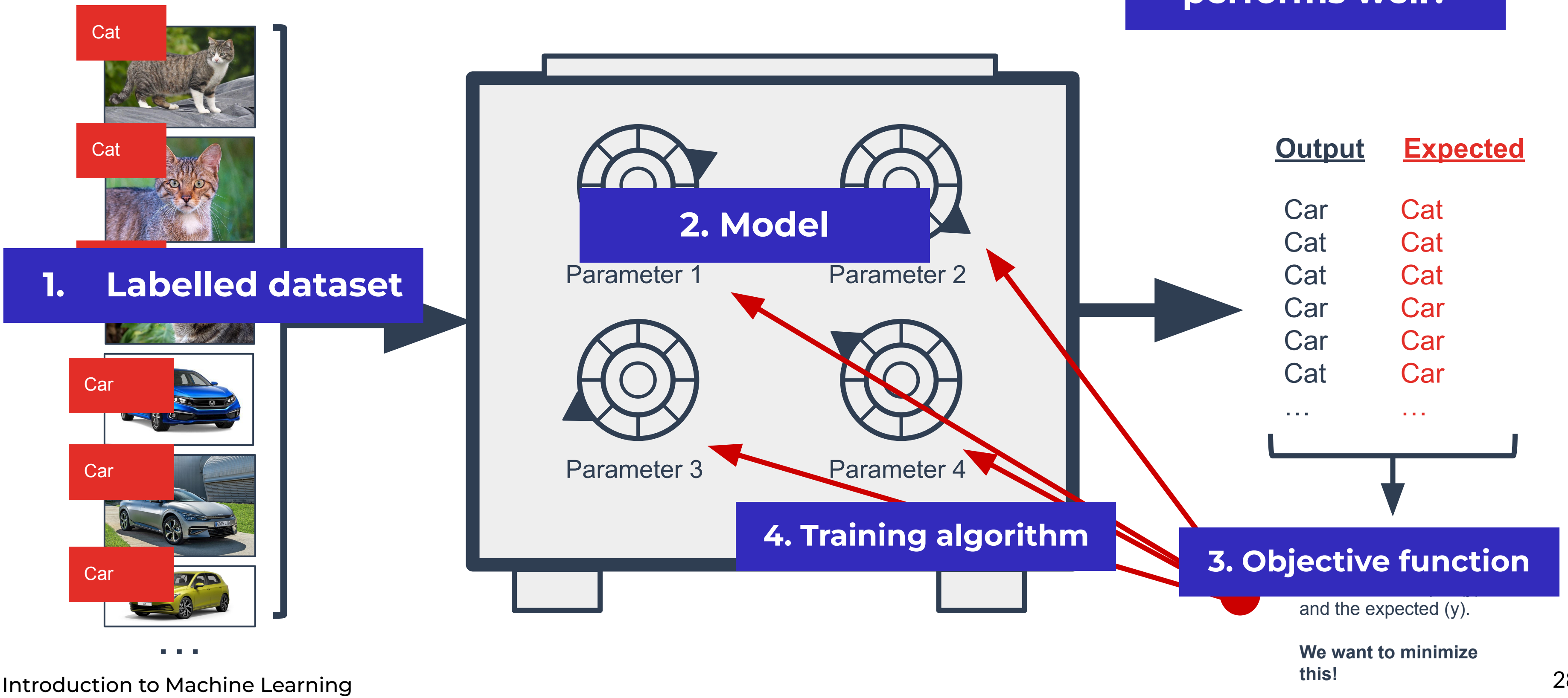


Objective function (L): Quantify the difference between the output (ŷ) and the expected (y).

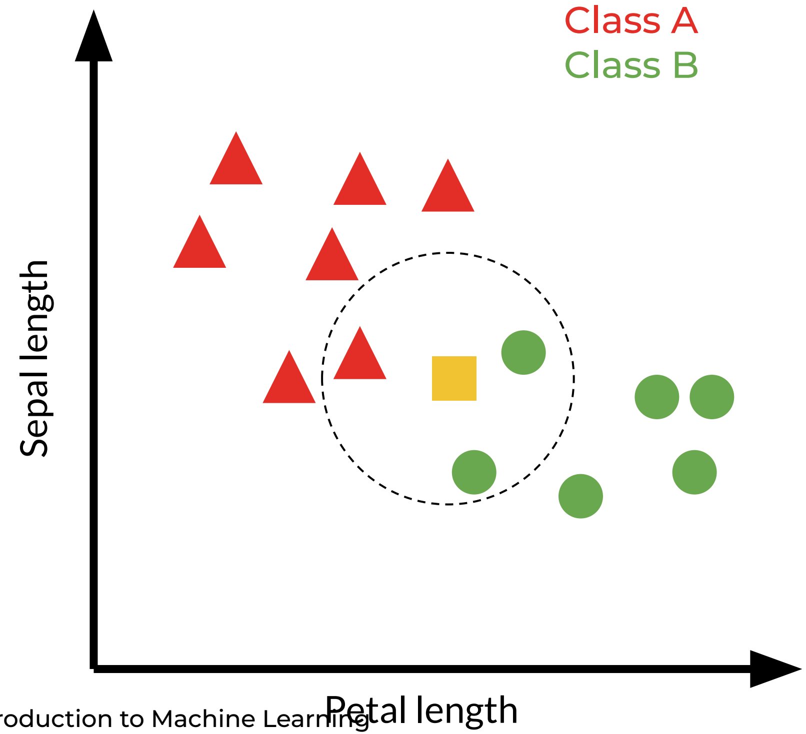
We want to minimize this!

Parametric models

5. How to we know that our model performs well?



Non-parametric methods

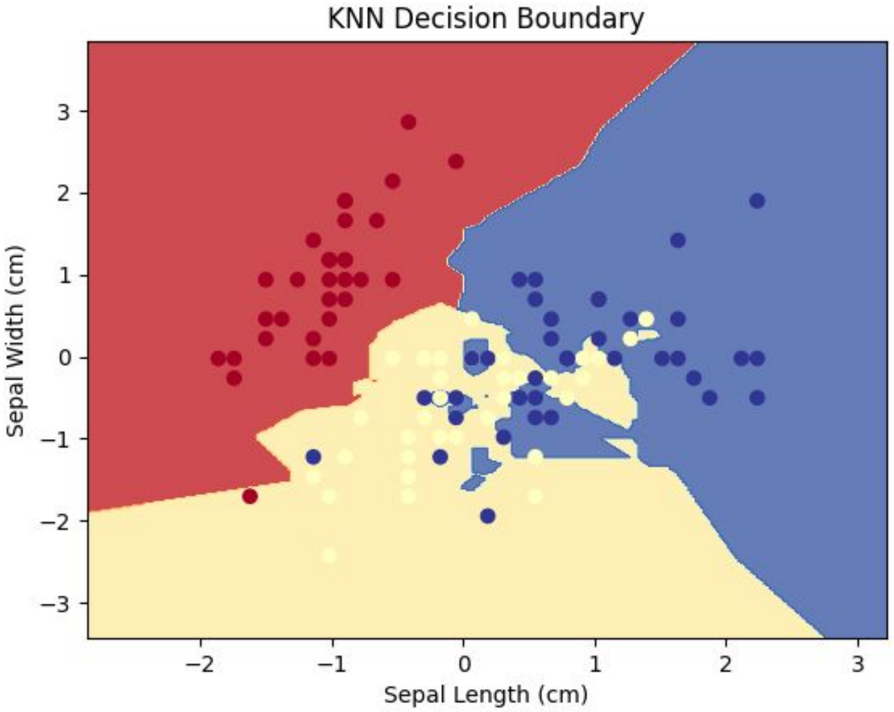


- Try to make assumptions about the data given the patterns observed from similar instances.
- Example: K-nearest neighbor (KNN)
  - Looks for similar training patterns for new instances.
  - Assumption: the most similar are most likely to have a similar result.
- What is similarity (features)? Distances: Euclidean, Manhattan.

Solving the IRIS with KNN

```
# Create a KNN classifier
k = 5  # Number of neighbors
clf = KNeighborsClassifier(n_neighbors=k)
clf.fit(X_train, y_train)
y_pred = clf.predict(X_test)
accuracy = accuracy_score(y_test, y_pred)
print(f'Accuracy on Test Set: {accuracy:.2f}')
# KNN Accuracy: 0.8
```

Solving the IRIS classification task



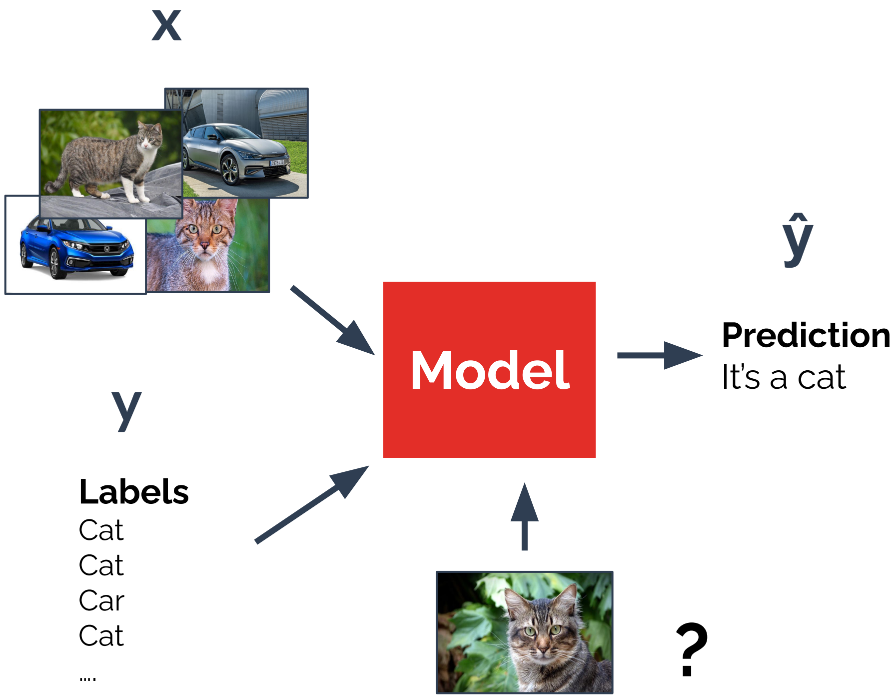
Learning methods

Supervised, unsupervised, self-supervised, reinforcement learning

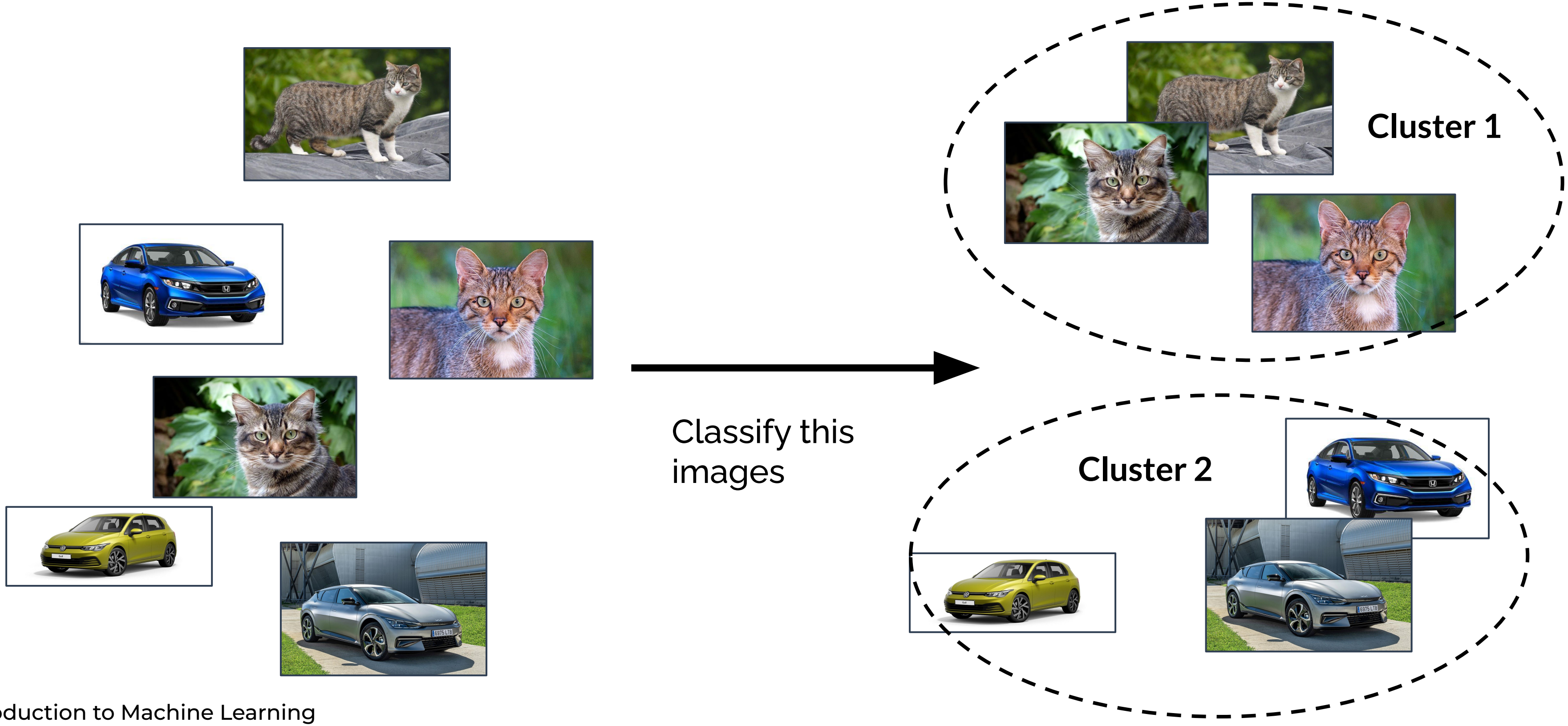
Supervised Learning

Learn a function that maps an input to an output based on examples of input-output pairs.

- Pairs of inputs (usually vectors) to and output value (target signal).
- Generalization error: we want to predict correctly on unseen examples!
- Problem: in a lot of cases you need humans to label the data !



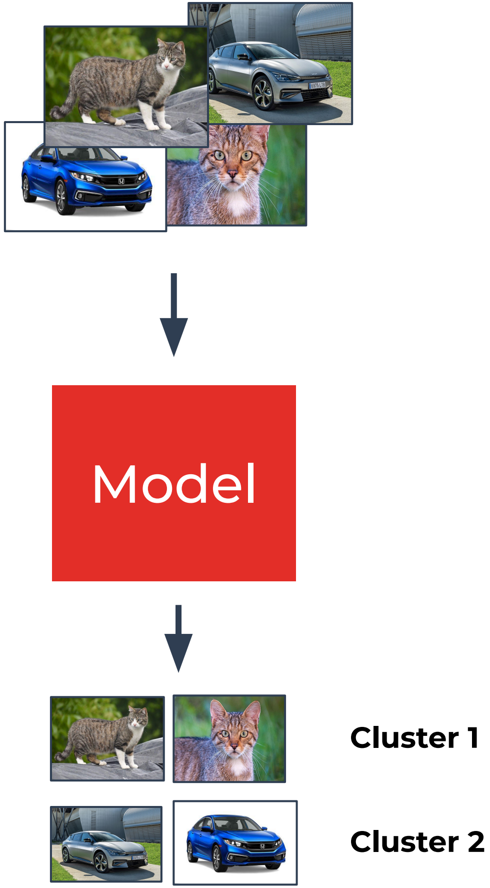
Unsupervised Learning



Unsupervised Learning

Learn the underlying patterns (as probability densities or extracted features) from untagged data.

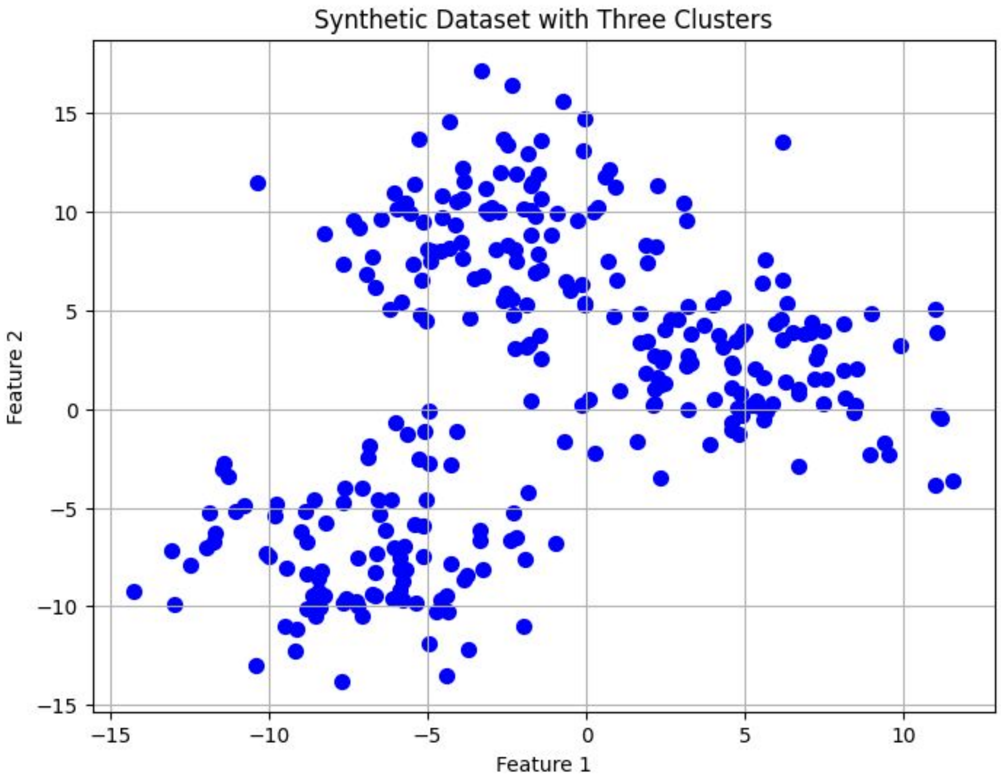
- We have only inputs (x) and no target signal (ŷ).
- Try to mimic that data and uses the error to correct itself.
- Try to represent the data in a simpler way and compare to reconstructed signal to the original.
- Group examples by similarity.



Example of unsupervised learning

```python
from sklearn.datasets import make_blobs
# Create a synthetic dataset with three clusters
X, y = make_blobs(
    n_samples=300,
    centers=3,
    random_state=42,
    cluster_std=3
)
```

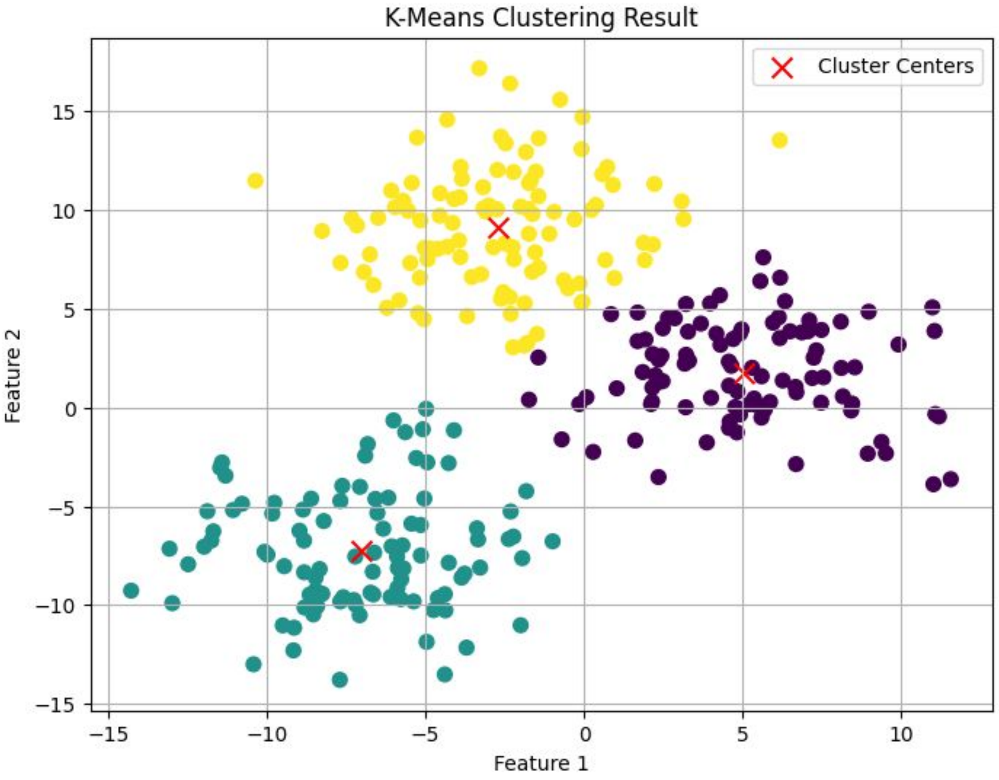
Example of unsupervised learning



Example of unsupervised learning

```python
from sklearn.cluster import KMeans


# Apply K-Means clustering
kmeans = KMeans(n_clusters=3, random_state=42)
kmeans.fit(X)
# Get cluster assignments and cluster centers
cluster_labels = kmeans.labels_
cluster_centers = kmeans.cluster_centers_
```

Example of unsupervised learning



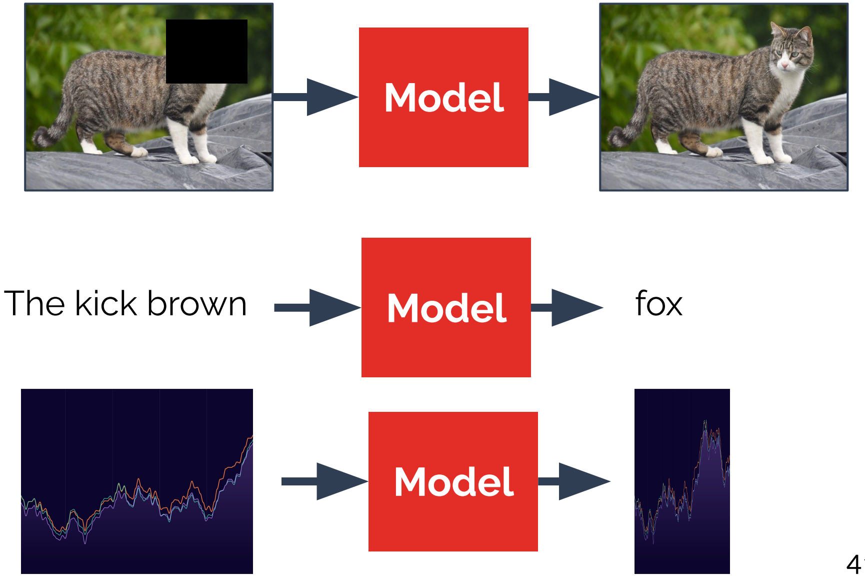
Example of unsupervised learning

https://colab.research.google.com/drive/15y1WjGUoDW03_dTrU-aWLWx9JZ03ClmN?usp=sharing

Self-supervised Learning

Key idea: the model trains itself to learn one part of the input from another part of the input.

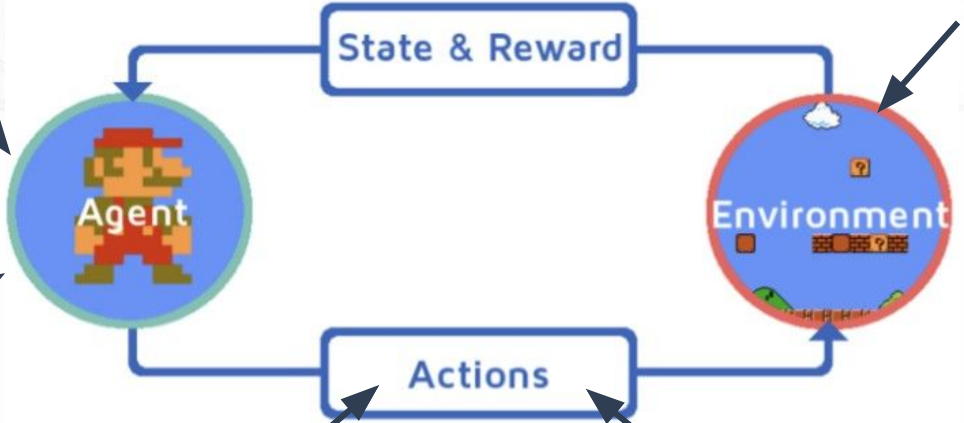
- Intermediate between supervised and unsupervised learning.
- Pseudo-labels: labels extracted from the data.
- Use the trained model on a related task.
- Linked to the brain: predictive coding.



Reinforcement Learning

The agent seeks to maximise the rewards receive from the environment.

The environment send back a new state (s) and a reward (r)



Based on Markov Decision Process (MDP)

The agent has a set of possible states (S).

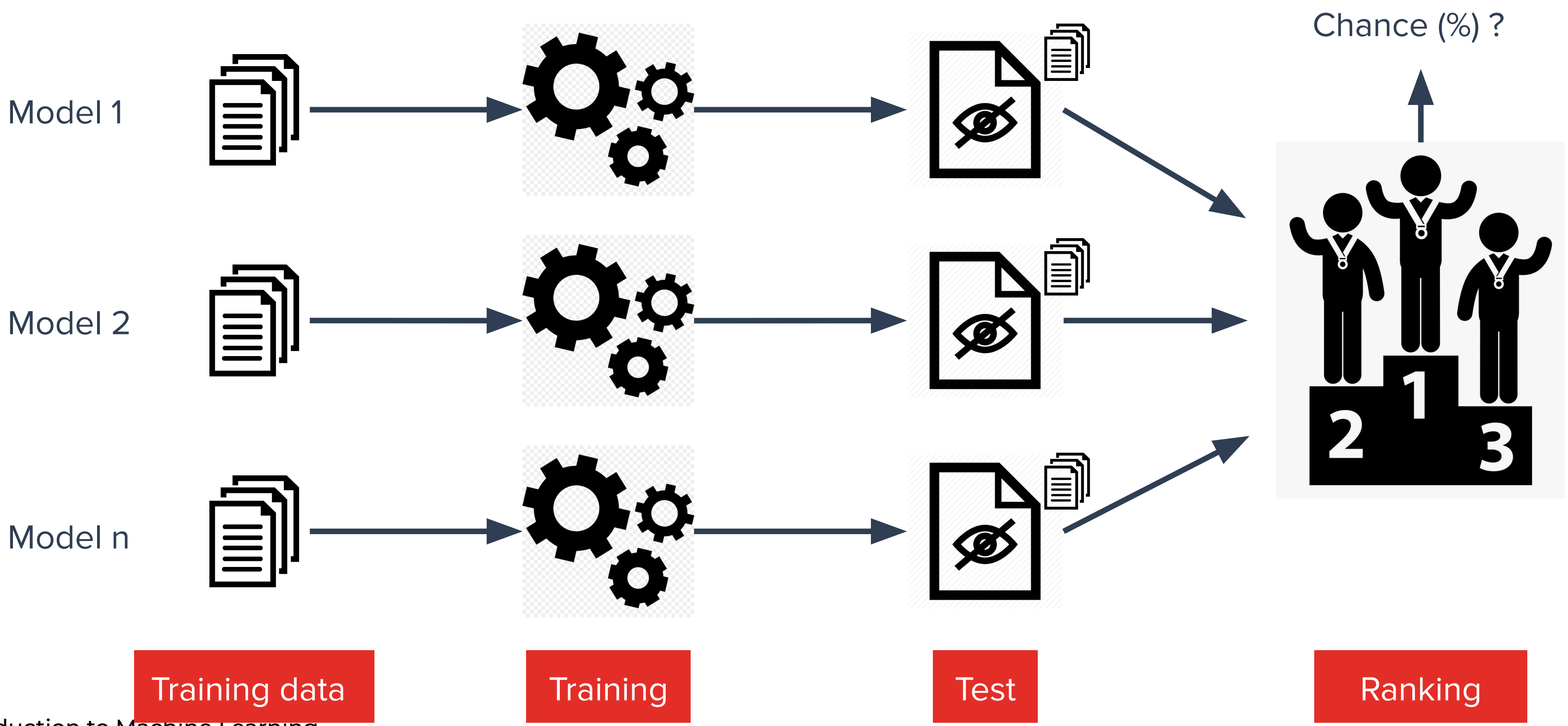
A set of possible actions depending on the state.

Policy
The agent choose an action (a) based on its state (s).

Methodology and applications

Model validation and selection

How select the best model ?



Methodology is the key

Train and test

Cross-validation

Ordered samples ?

- What's the definition of a good model?
- There is a lot of ways to measure the performance of a model.
- Performance (right predictions)
- Efficiency (fast predictions)
- Ethical (fair predictions)
- Economical balance (cost of false predictions)
- What data to choose to test the model? How much?
- Dependencies in training and test samples (shuffle first?).

Cross validation

```python
from sklearn.model_selection import cross_val_score
# Perform k-fold cross-validation (e.g., 5-fold)
num_folds = 5
scores = cross_val_score(
    regression_model,
    X, y,
    cv=num_folds,
    scoring='neg_mean_squared_error'
)
```

Cross validation

```
# Initialize the models
model_lr = LogisticRegression(max_iter=1000)
model_svm = SVC()

# Perform 5-fold cross-validation for each model
cv_scores_lr = cross_val_score(model_lr, X, y, cv=5)
cv_scores_svm = cross_val_score(model_svm, X, y, cv=5)

# Logistic Regression CV Scores: [0.96 1. 0.93 0.96 1.0]
# SVM CV Scores: [0.96 0.96 0.96 0.93 1.]
```

Cross validation

```
# Perform a paired t-test to compare the models' performance
t_stat, p_value = ttest_rel(cv_scores_lr, cv_scores_svm)

# t-statistic: 0.53
# p-value: 0.6213

# There is 62% of chance that the difference between SVM and
logistic regression is due to randomness, that’s a lot !
```

Example of unsupervised learning

https://colab.research.google.com/drive/1cE7ZxqmMqBZqoH
gTqjYcWsi5ALqeH1zI?usp=sharing