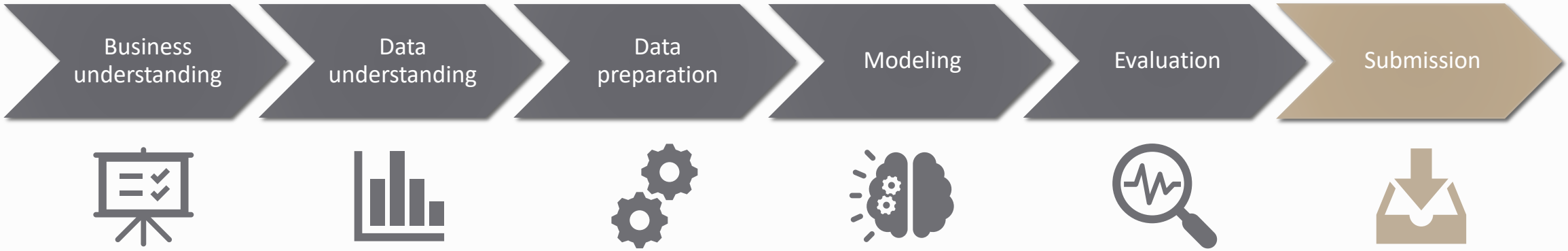
Parasite Eggs Identification

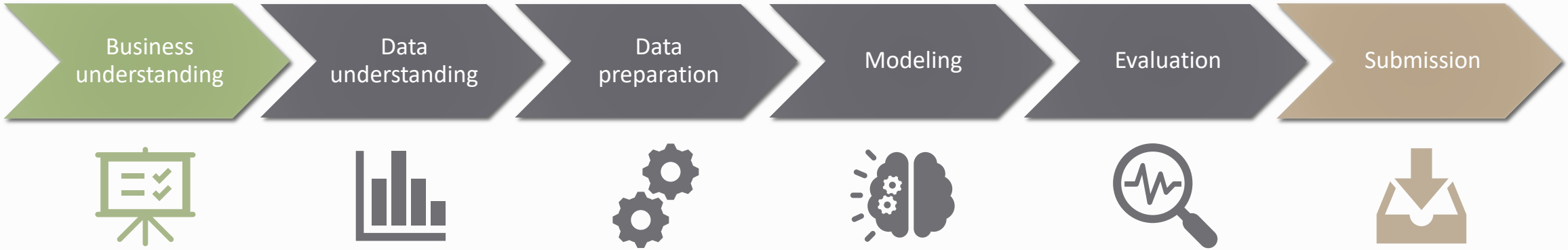
Unina – Data Mining 20/21 – Mini Contest N°3

Fabio d’Andrea – M63000989



Data Mining Process

Business understanding

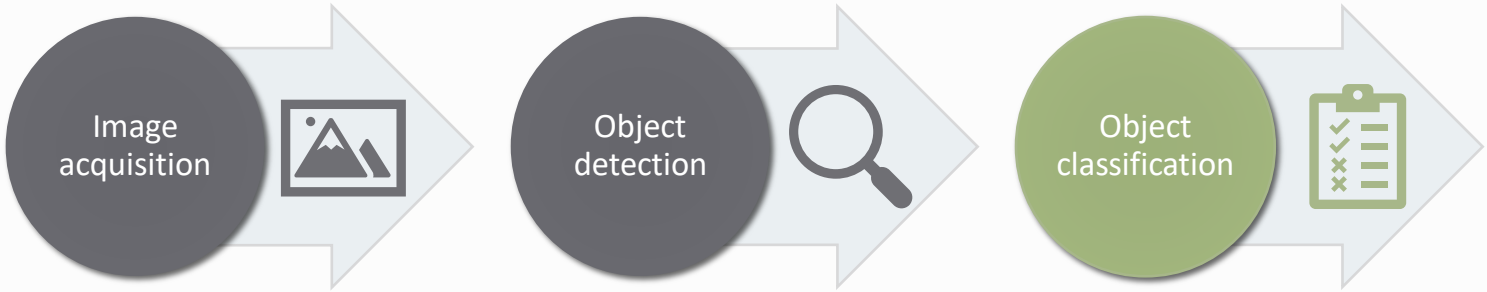


Data Mining Process

Business understanding

Business Understanding

Determine whether a parasite egg appears in a certain image



Data Mining Tools

Software tools used to face the problem

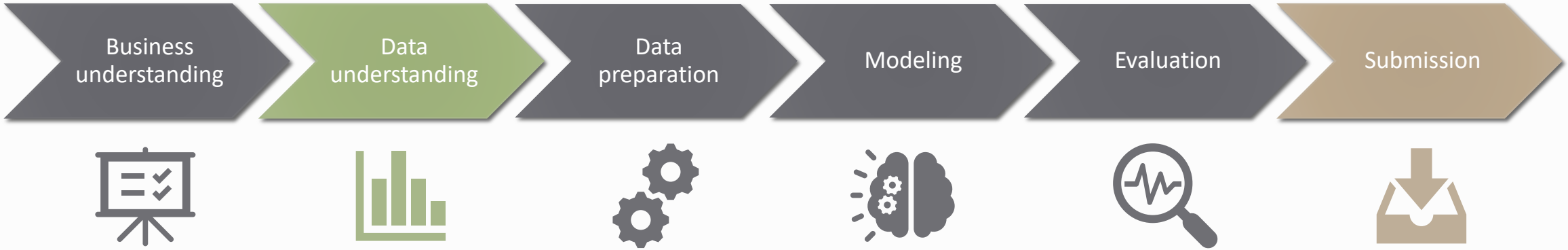
Python

Pandas

Matplotlib

PyTorch

Colab

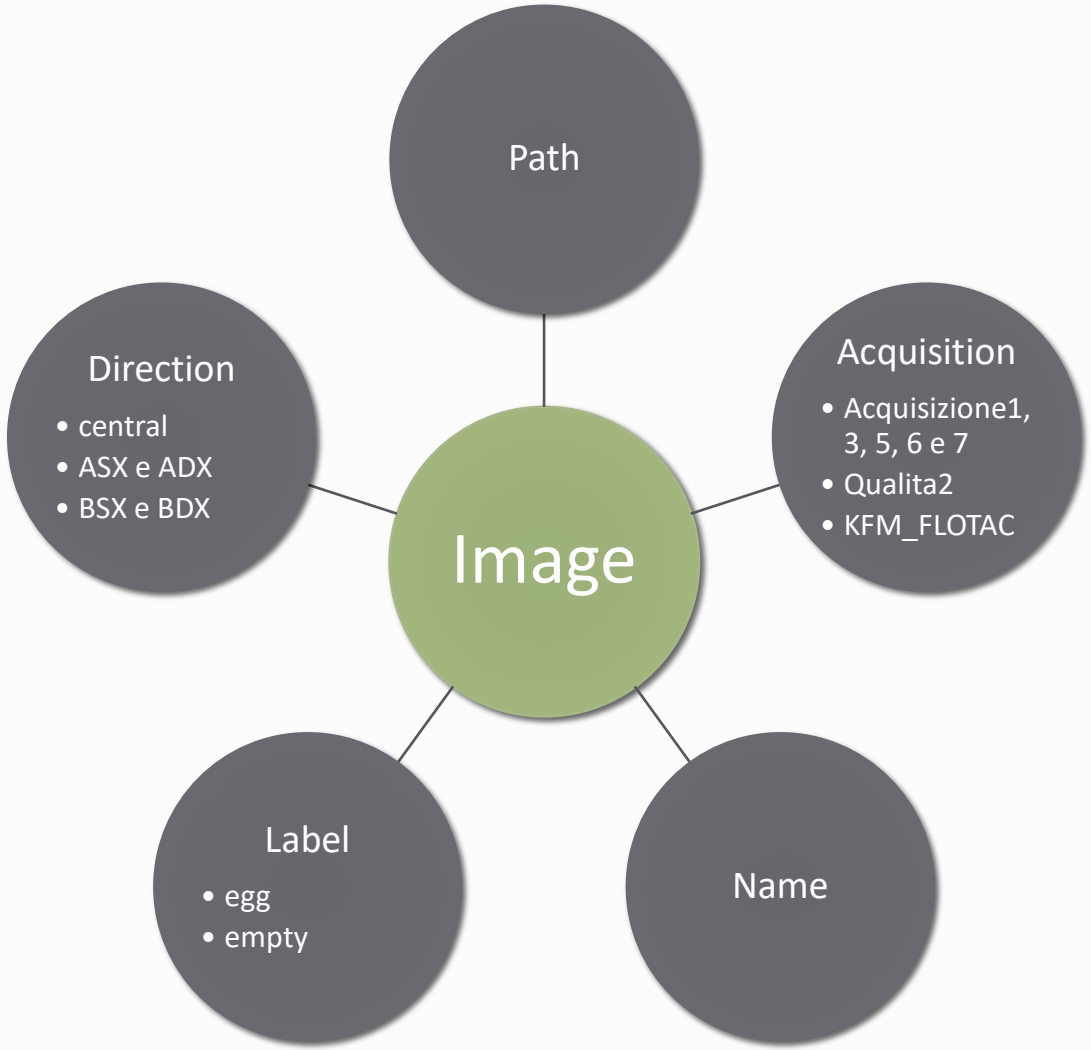


Data Mining Process

Data understanding

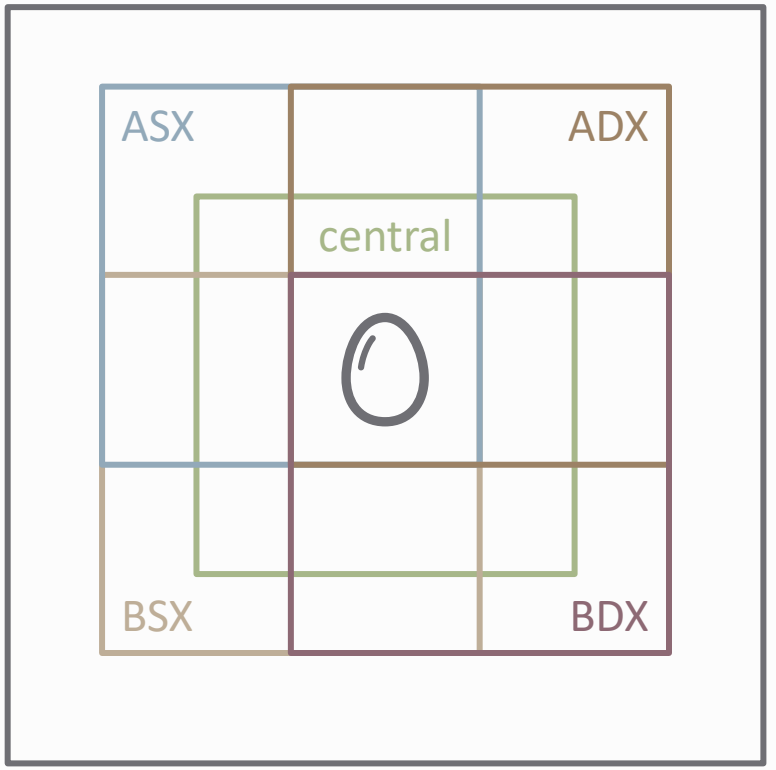
Data Understanding

Each training image is characterized by different attributes



Directions

5 images are extracted from the same egg sample by cropping the image from different directions



Data Understanding

Extract some useful information from the path of the images

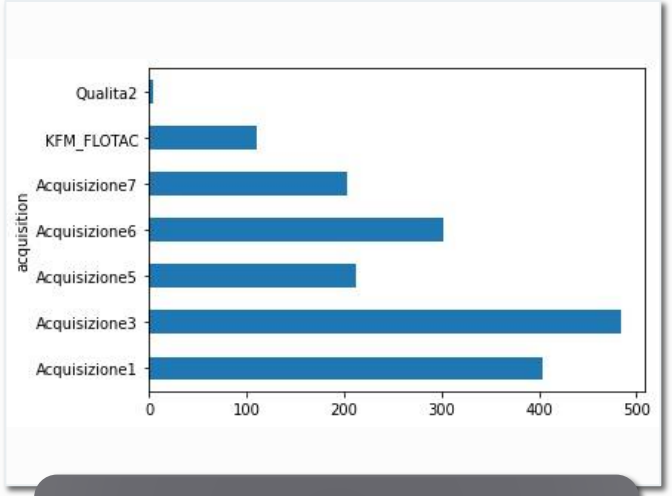
	path	acquisition	name	direction	label
0	/content/dataset/train/empty/Acquisizione5_mic...	Acquisizione5	micro_02_04	central	empty
0	/content/dataset/train/empty/Acquisizione1_mic...	Acquisizione1	micro_06_08	central	empty
0	/content/dataset/train/empty/Acquisizione7_mic...	Acquisizione7	micro_12_05	central	empty
0	/content/dataset/train/empty/Acquisizione7_mic...	Acquisizione7	micro_03_00	central	empty
0	/content/dataset/train/empty/Acquisizione3_mic...	Acquisizione3	micro_07_06	central	empty

Build a **dataframe** containg the information extracted

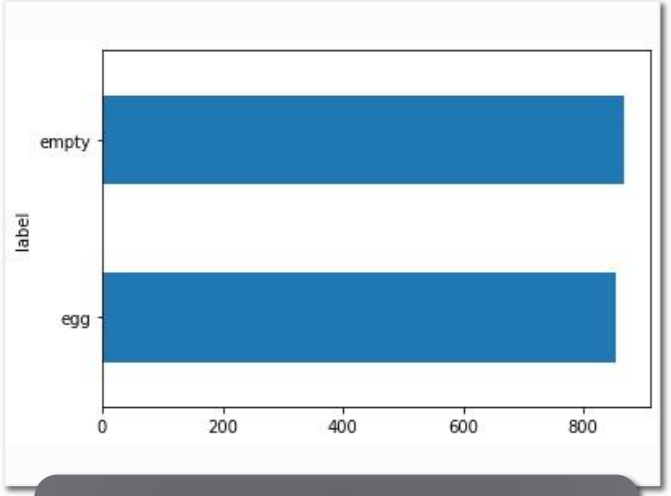
	path	acquisition	name	direction	label
count	1722	1722	1722	1722	1722
unique	1722	7	373	5	2
top	/content/dataset/train/empty/Acquisizione7_mic...	Acquisizione3	micro_00_03_1	central	empty
freq	1	484	15	1038	867

Compute some basic descriptive statistics about data

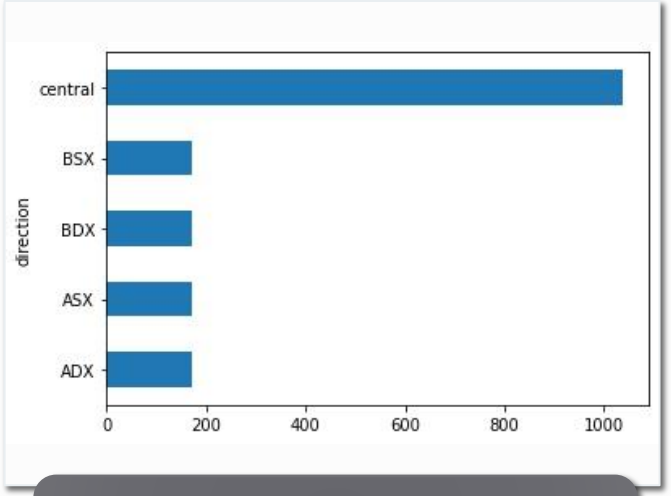
Data Understanding



Different acquisitions have a different number of images

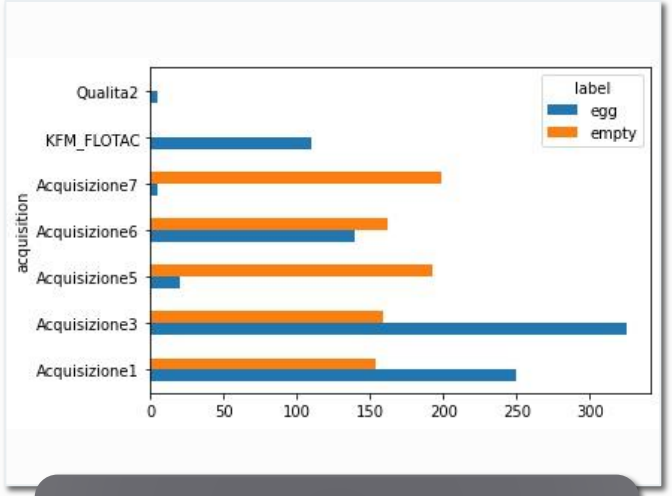


Thanks to multiple directions, training set is balanced in term of class

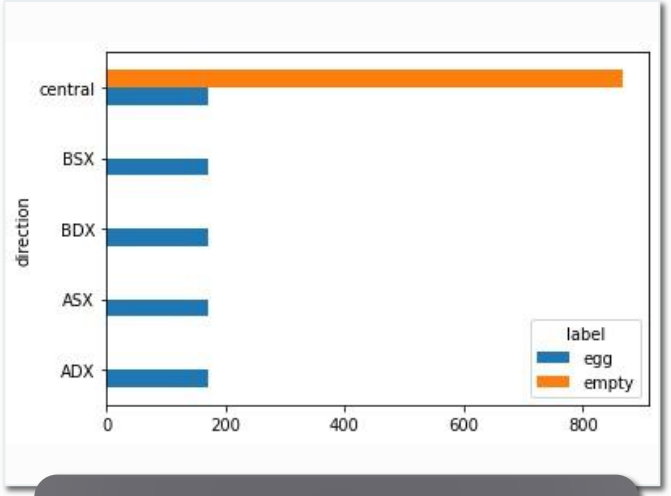


For each egg image there are all the five directions

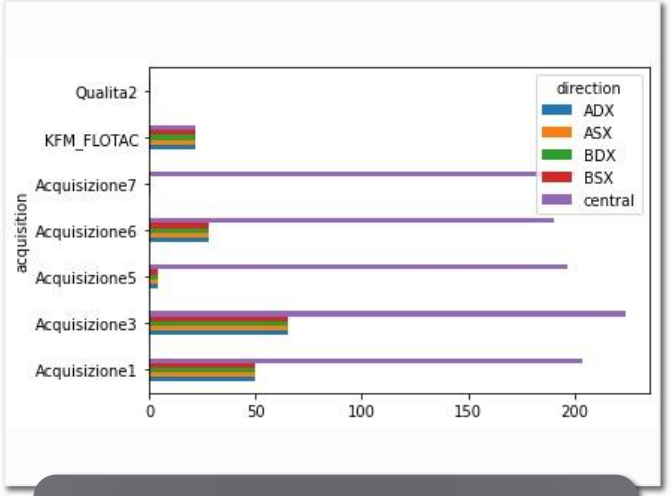
Data Understanding



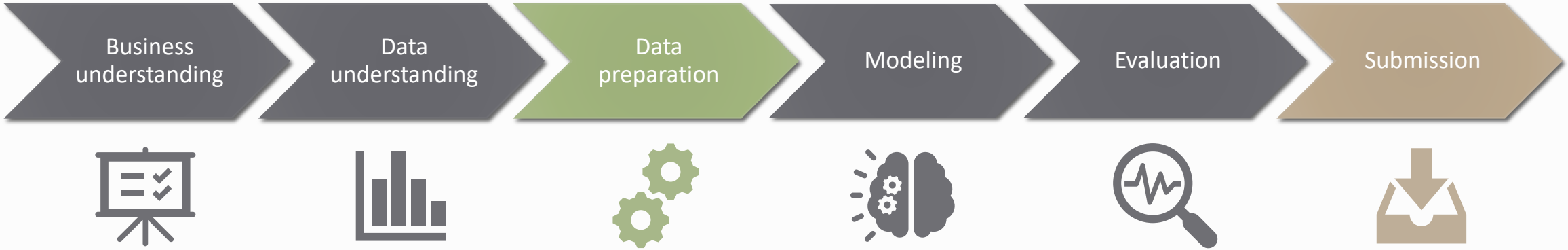
Class distribution for different acquisition



Different directions are used only for egg images



Different directions of the same image refer to the same acquisition

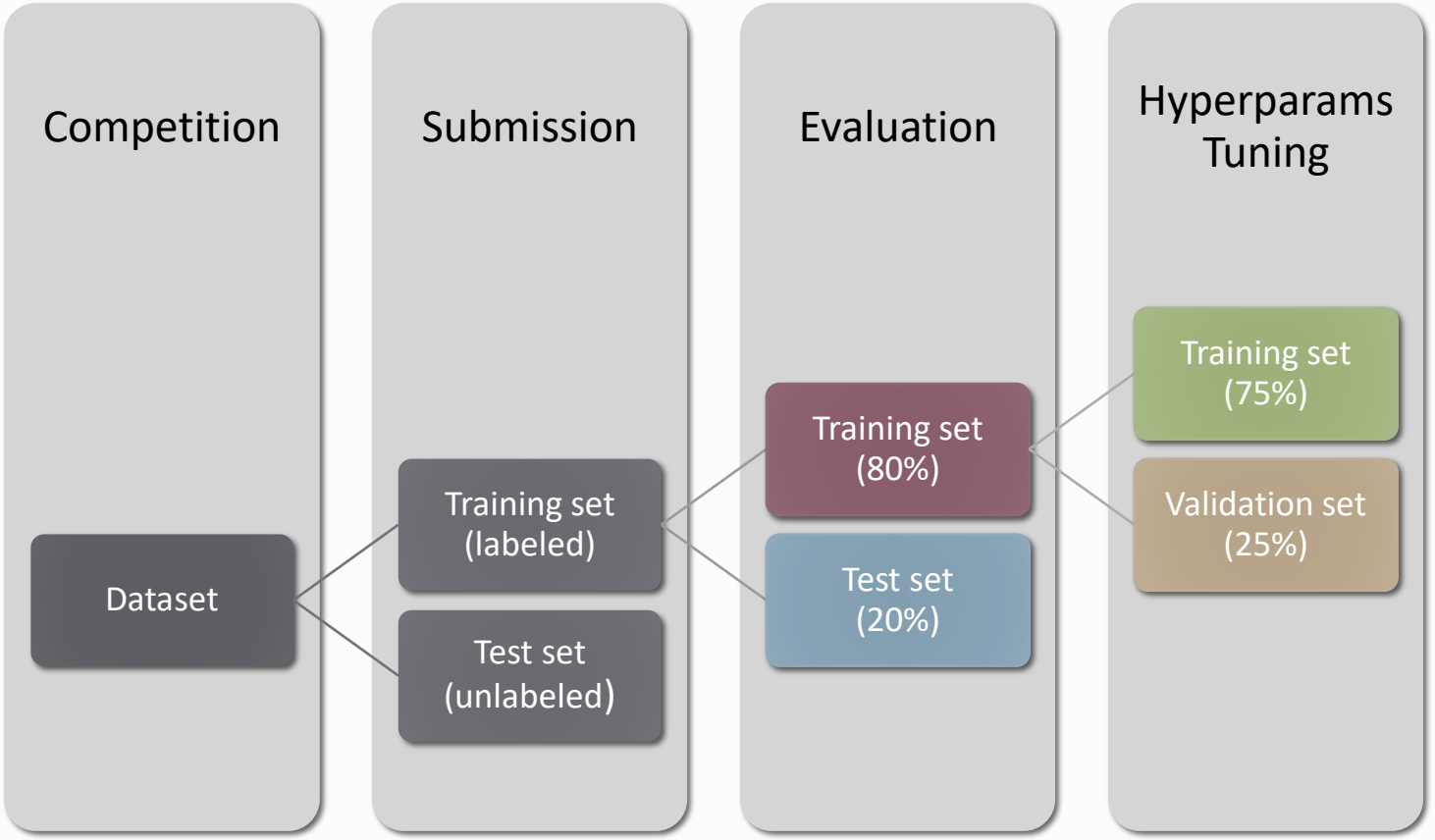


Data Mining Process

Data preparation

Data Preparation

Apply **stratified holdout**, ensuring that each subset is balanced in term of class

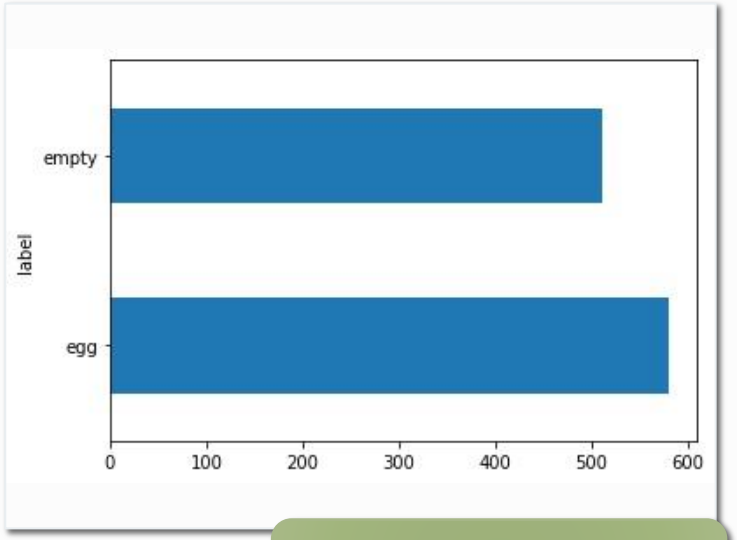


Data Preparation

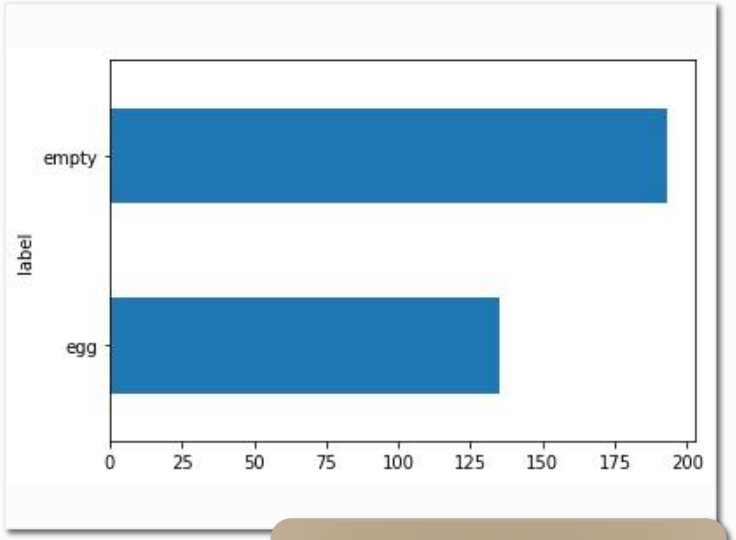
To be as independent as possible, the three subsets are made up of different acquisitions

Training set (60%)	Validation set (20%)	Test set (20%)
• Acquisizione1 • Acquisizione3 • Acquisizione7	• Acquisizione5 • KFM_FLOTAC • Qualita2	• Acquisizione6

Data for Hyperparameters Tuning

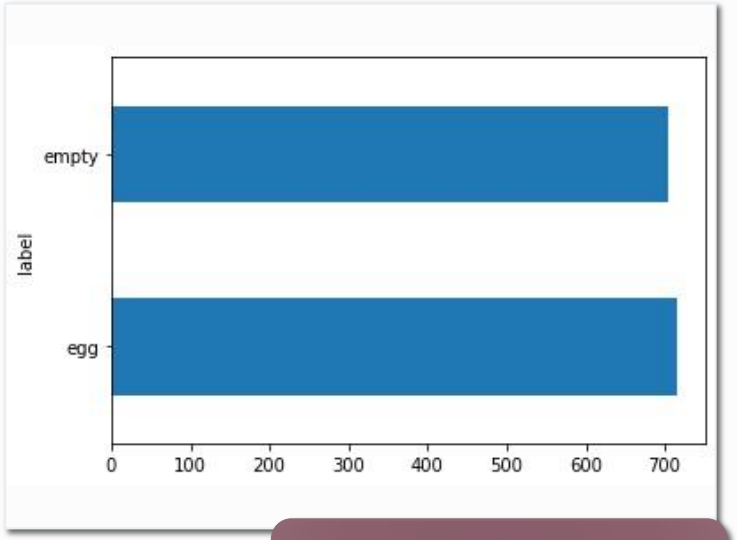


Training set

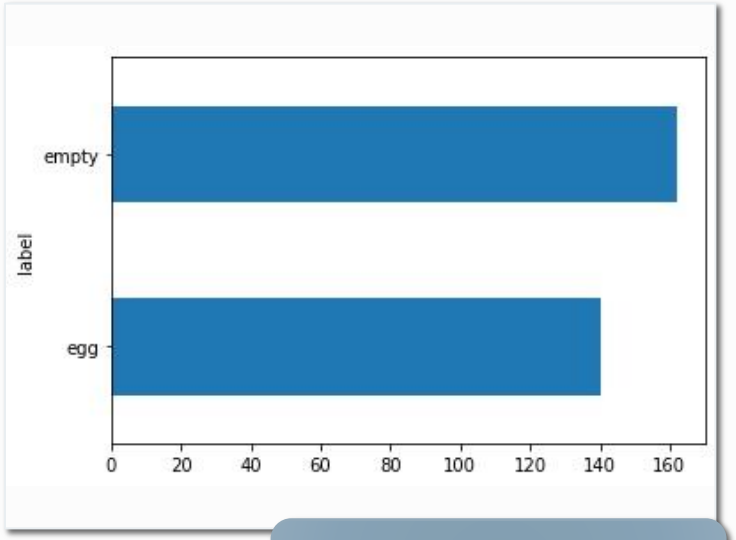


Validation set

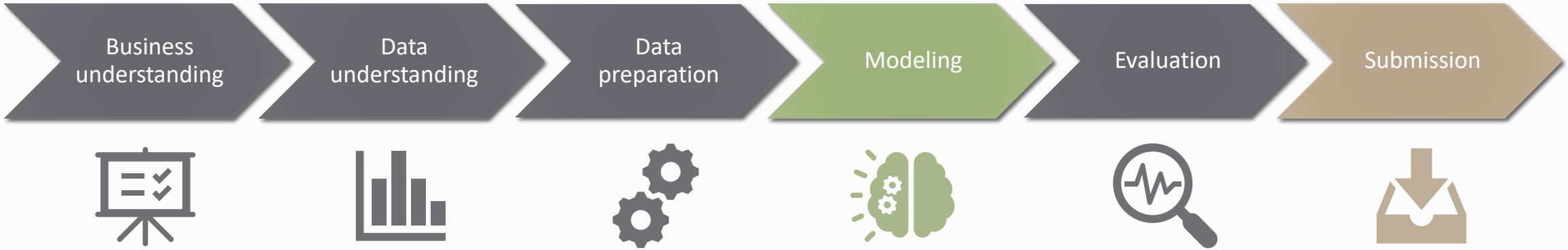
Data for Evaluation and Model Selection



Training set



Test set



Data Mining Process

Modeling

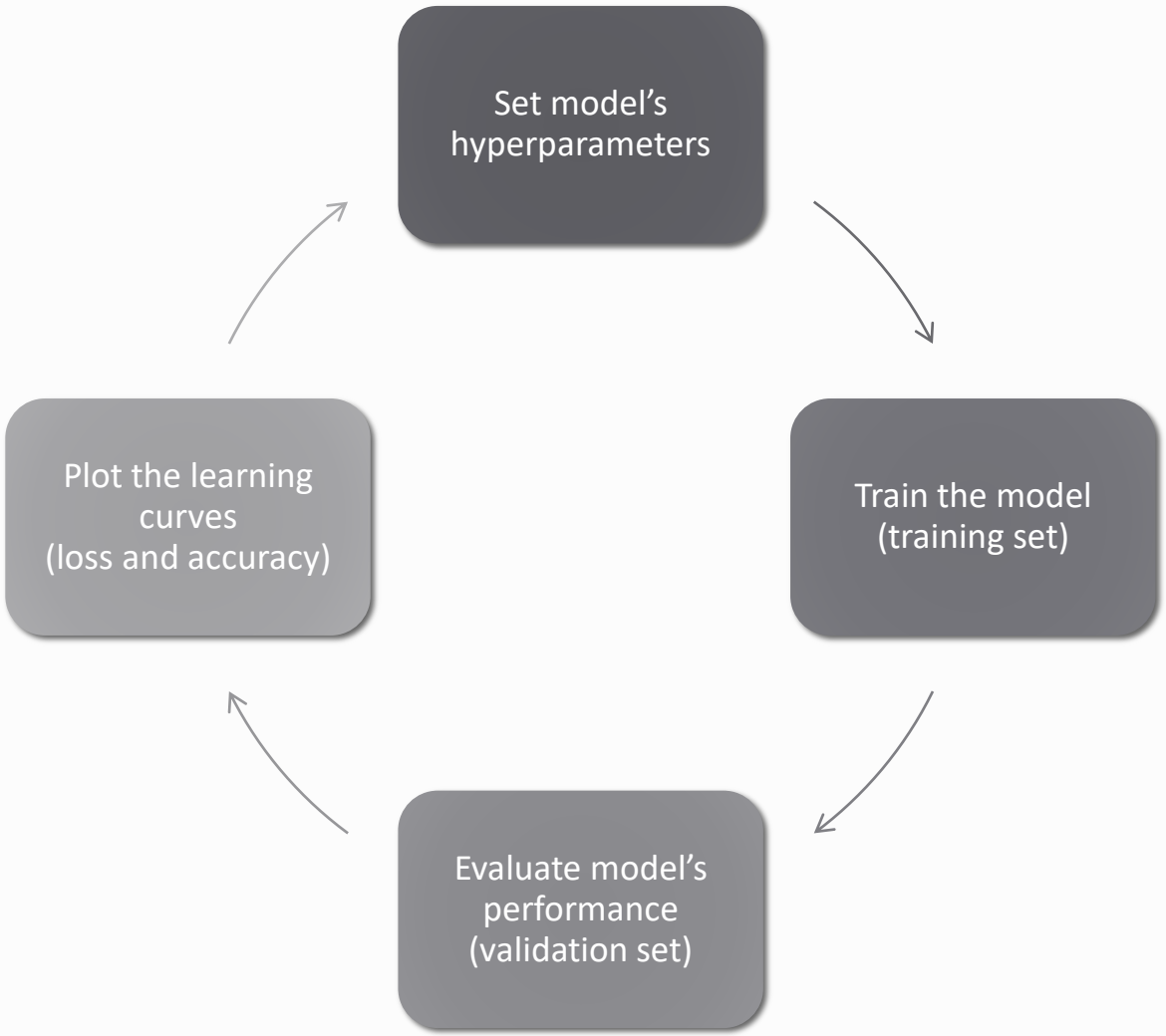
Modeling

Consider three different pretrained Convolutional Neural Networks (CNNs), provided by PyTorch

AlexNet	VGG11_bn	GoogLeNet
• 5 convolutional layers • Dropout • 3 dense layers	• 8 convolutional layers • Batch normalization • Dropout • 3 dense layers	• 21 convolutional layers • Inception • Batch normalization • Dropout • 1 dense layer

Modeling

Repeat the whole process for each neural network

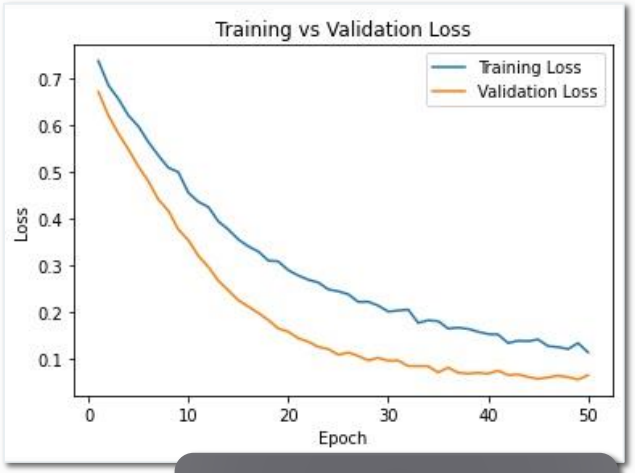


Hyperparameters Tuning

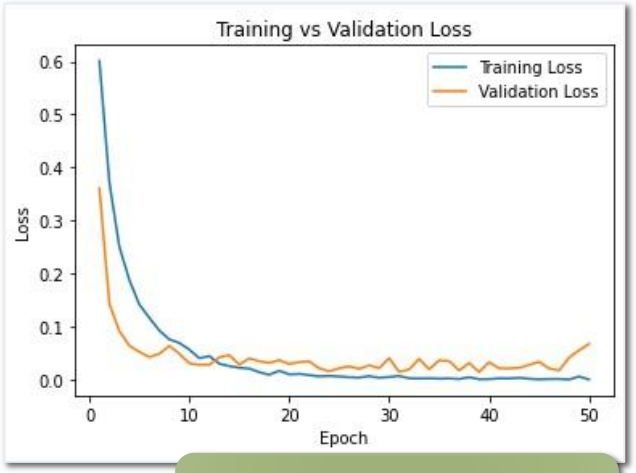
Tune the hyperparameters evaluating model's performance on validation set

Hyperparameter	Description
Trained layers	Layers whose parameters are learned
Batch size	Size of mini batches
Number of epochs	Number of training epochs
Loss function	Function to minimize during training
Optimizer	Optimization algorithm used for training
Learning rate	Learning rate used by the optimization algorithm
Data augmentation	Transformation applied to training images

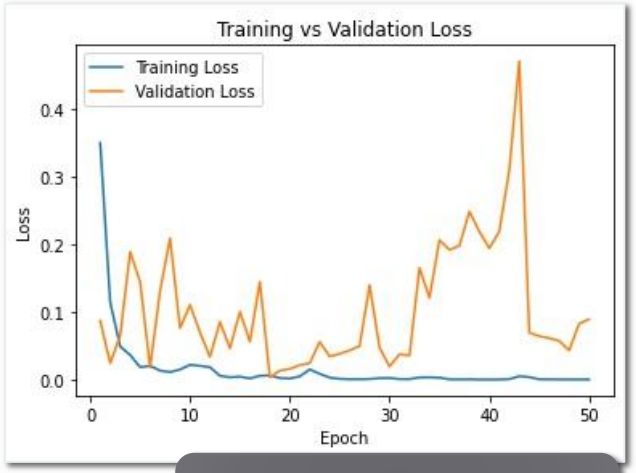
Ex: Tuning the Learning Rate (VGG11_bn)



Low learning rate

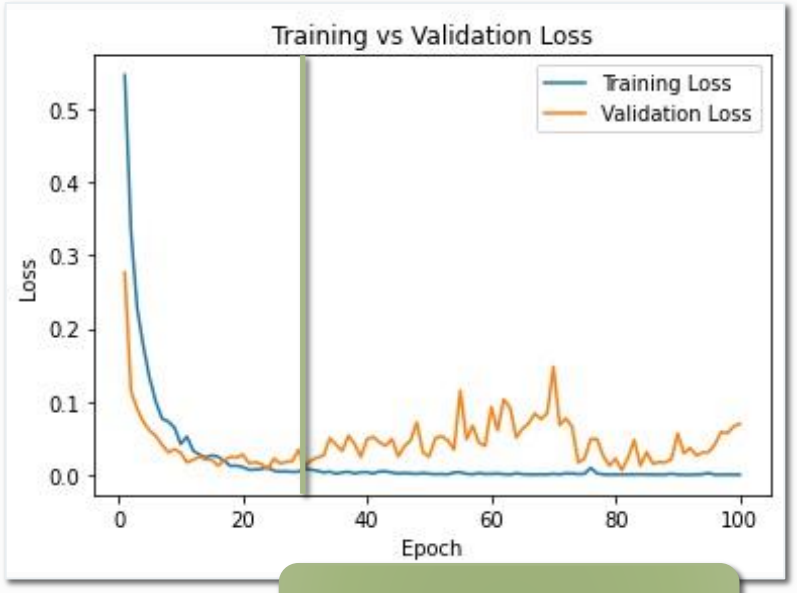


Good learning rate



High learning rate

Ex: Tuning the Epochs (GoogLeNet)

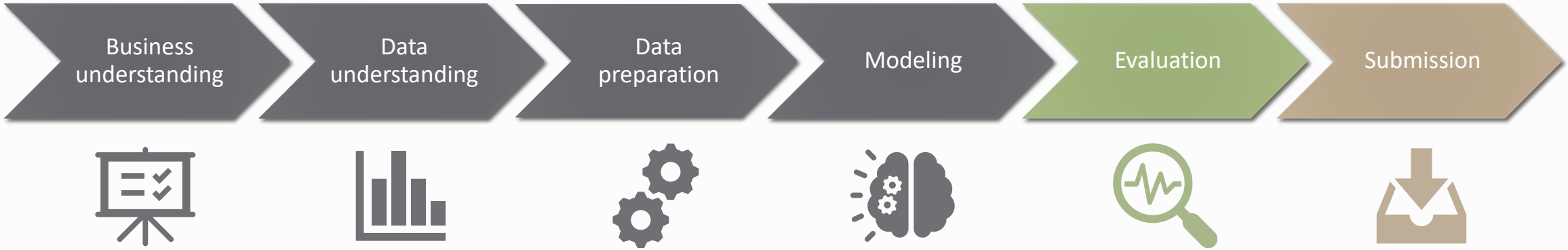


Final Hyperparameters

AlexNet	
Hyperparam	**Value**
Trained layers	Fully-connected
Batch size	32
Number of epochs	30
Loss function	Cross entropy
Optimizer	Adam
Learning rate	1e-5
Data augmentation	Flip and rotation

VGG11_bn	
Hyperparam	**Value**
Trained layers	Fully-connected + last convolutional
Batch size	32
Number of epochs	30
Loss function	Cross entropy
Optimizer	Adam
Learning rate	1e-5
Data augmentation	Flip

GoogLeNet	
Hyperparam	**Value**
Trained layers	All
Batch size	32
Number of epochs	30
Loss function	Cross entropy
Optimizer	Adam
Learning rate	1e-5
Data augmentation	Flip

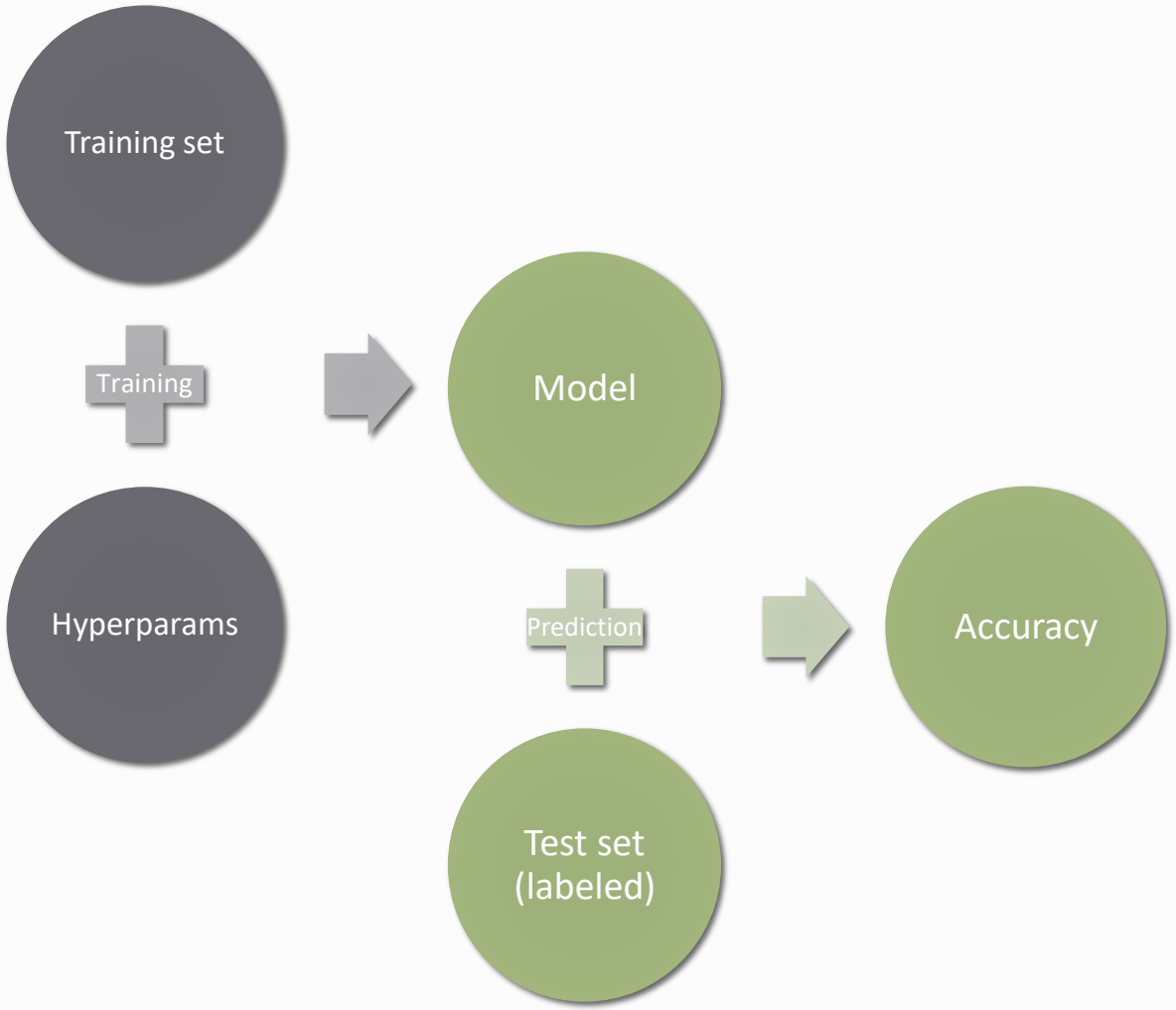


Data Mining Process

Evaluation

Evaluation

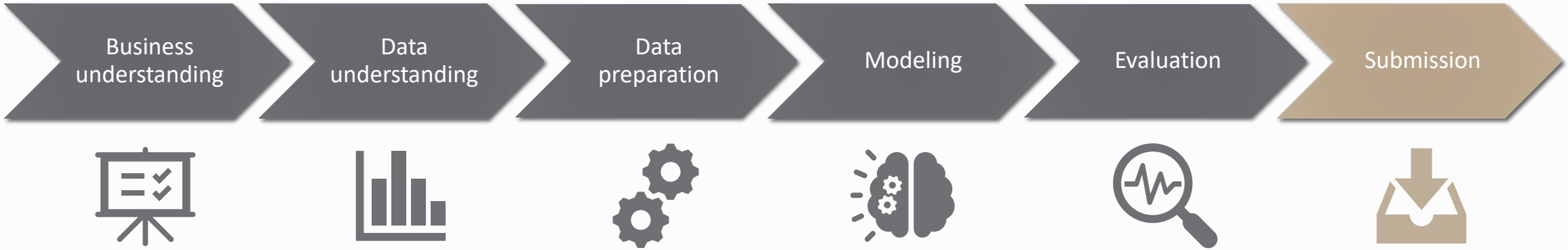
Evaluate models' performance on an **independent** test set



Model Selection

Choose the model that gains the best performance on the test set

	AlexNet	VGG11_bn	GoogLeNet
Accuracy (test set)	0.79139	0.95364	0.97019
Accuracy (public leaderboard)	-	-	-
Accuracy (private leaderboard)	-	-	-

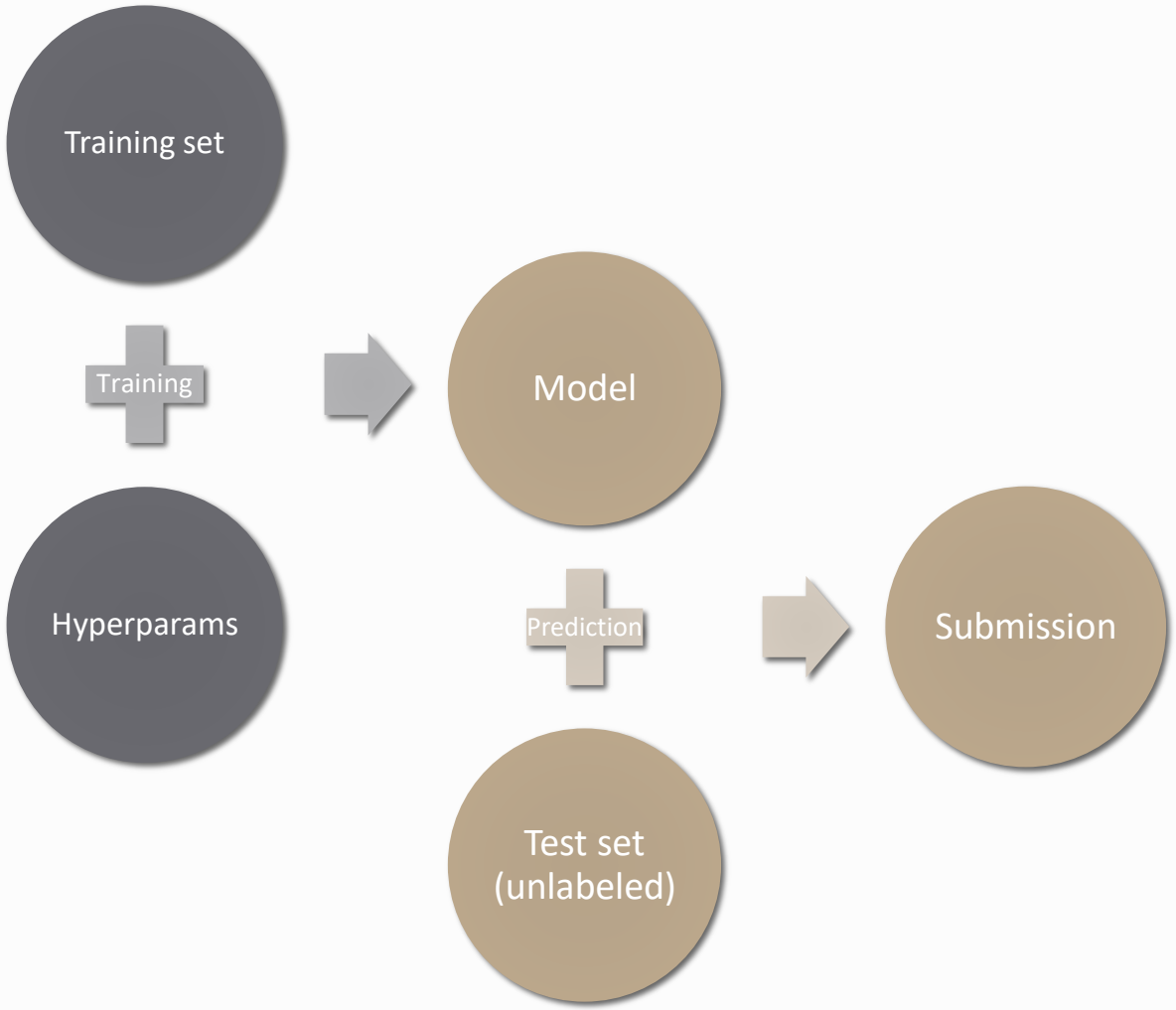


Data Mining Process

Submission

Submission

Retrain the model using the whole training set and predict the labels for the instances of the original test set



Final Score

The model with better performance on the test gets also the best score on Kaggle

	AlexNet	VGG11_bn	GoogLeNet
Accuracy (test set)	0.79139	0.95364	0.97019
Accuracy (public leaderboard)	-	0.98058	0.98058
Accuracy (private leaderboard)	-	0.95748	0.98897

Thanks for your attention!