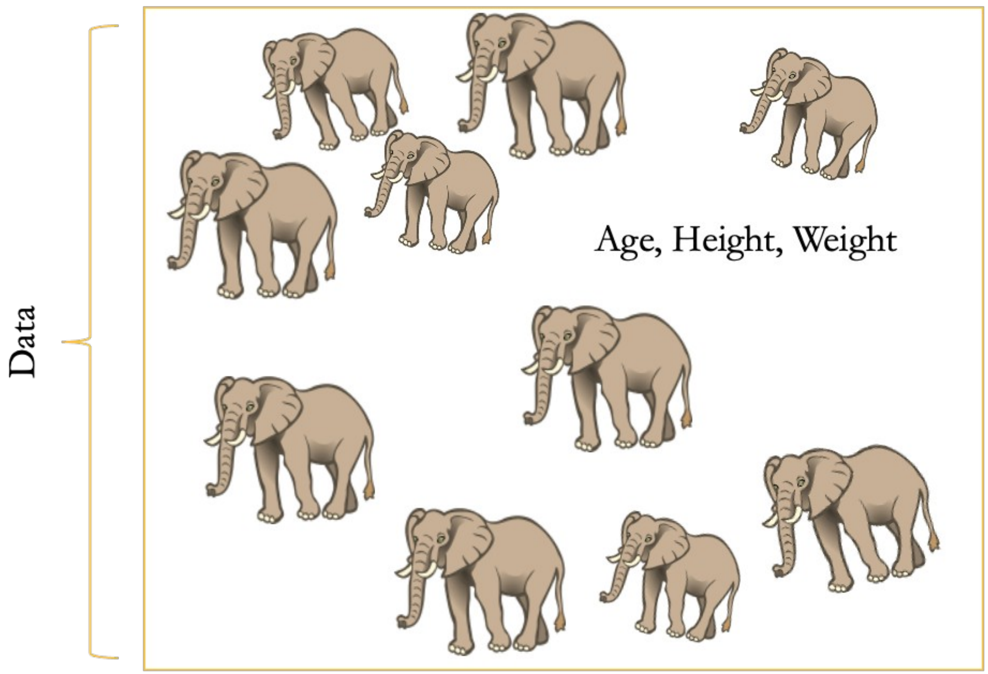
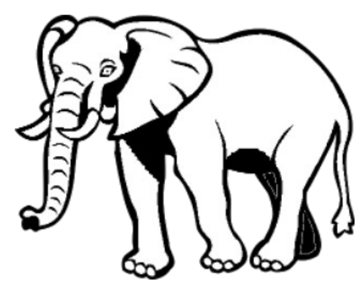
Model Validation - Cross Validation & Model Validation w/ Categorical Response

Cross Validation Motivation - Elephant Example

Say we want to build a model to predict an elephant's weight based on two predictors – age and height.

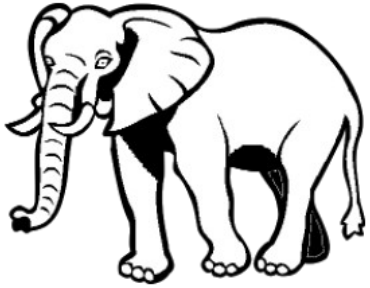


We will want to know how well our model will perform if we use it to predict the weight of an elephant we haven't seen before.



Model Validation

New Elephant
Age = 10 Years, Height = 8’



$\hat{f}(height, age)$

MODEL

Prediction
$\hat{y}$

Prediction

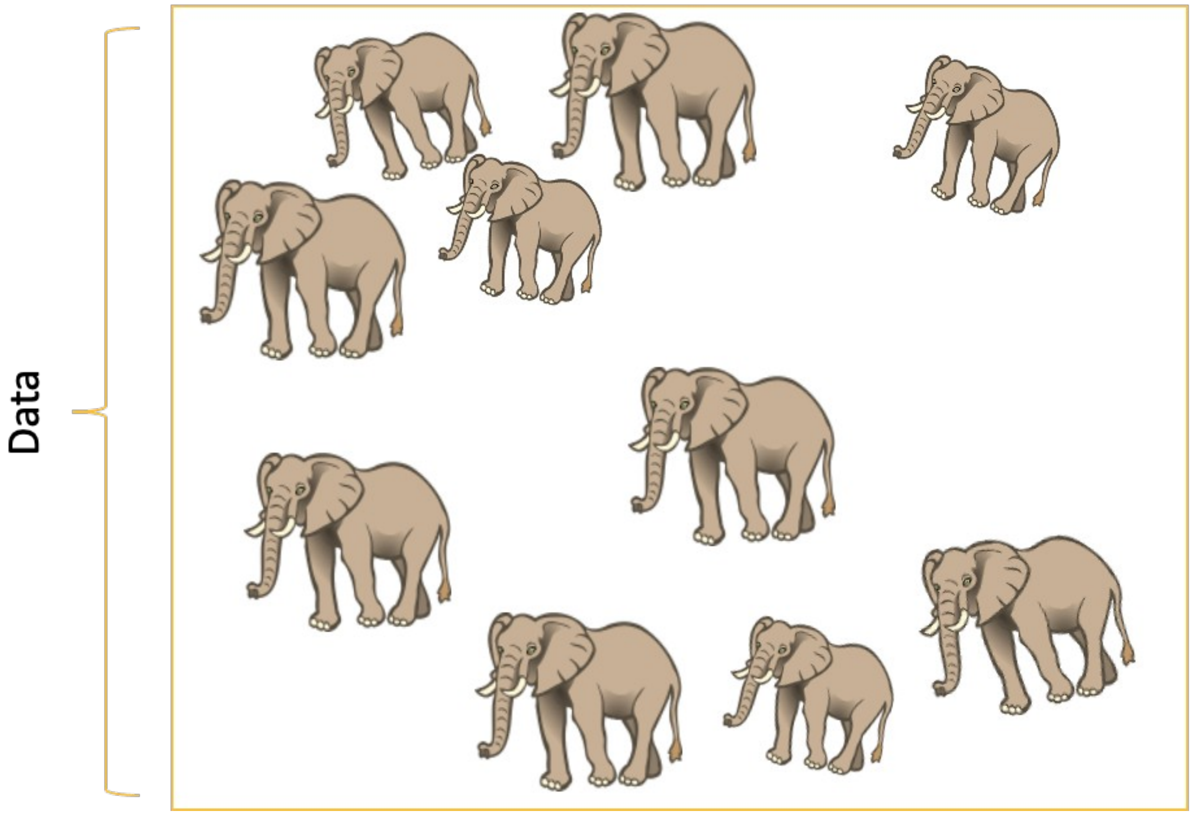
$\hat{y} = 4000\ lb$

Actual Value

$y = 3988\ lb$

Model validation is the process of assessing a model’s performance on **new data** by comparing the model’s prediction to the actual value.

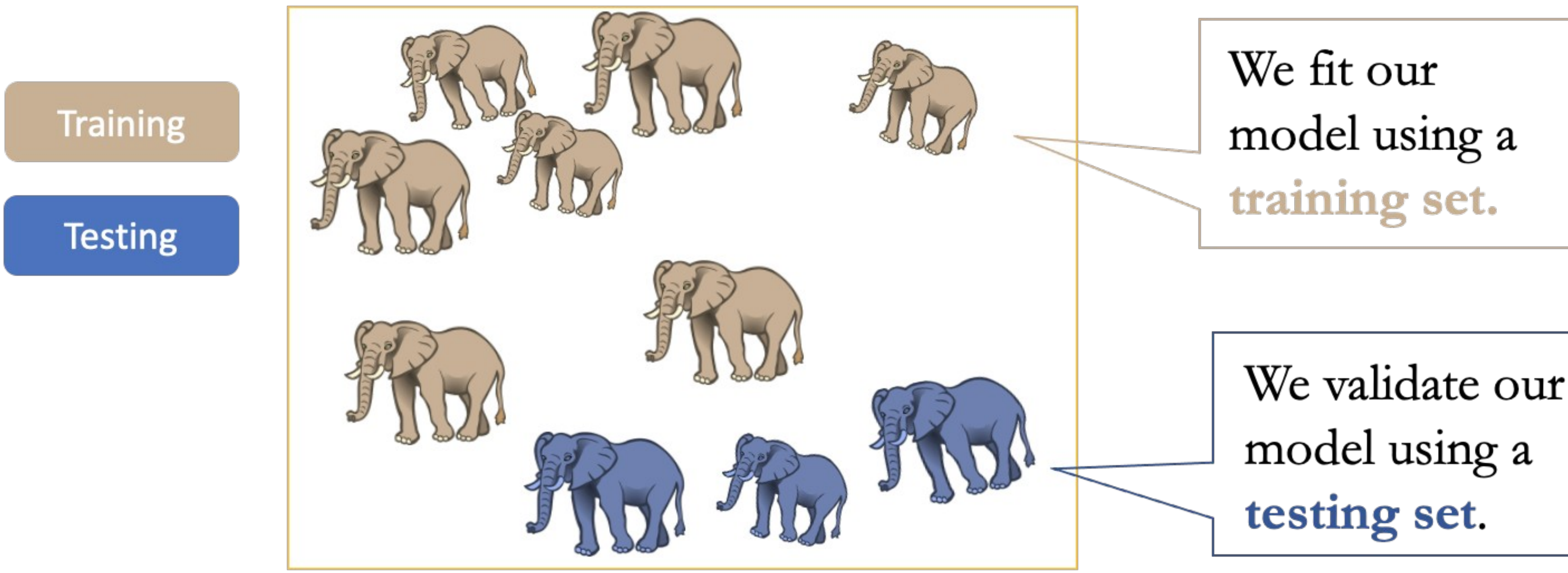
How Do We Validate Our Model?



How can we **build** & **validate** our model using these 10 elephants?

Cross Validation (CV)

To assess how our model will perform with new data, we can split our data into training and testing sets. This is a single-run **cross validation**.

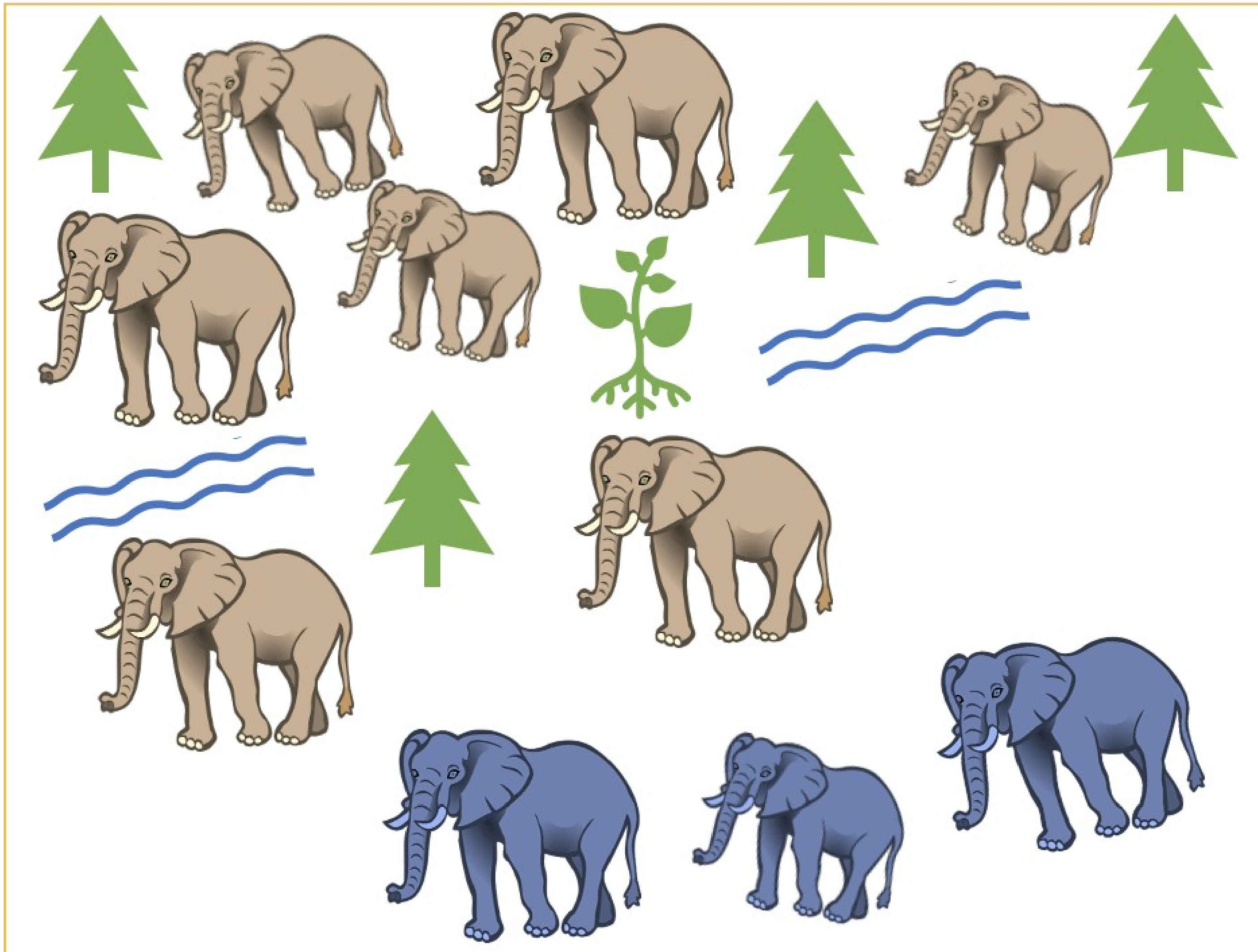


Cross validation (CV) allows us to assess our model's predictive ability using a "new" data.

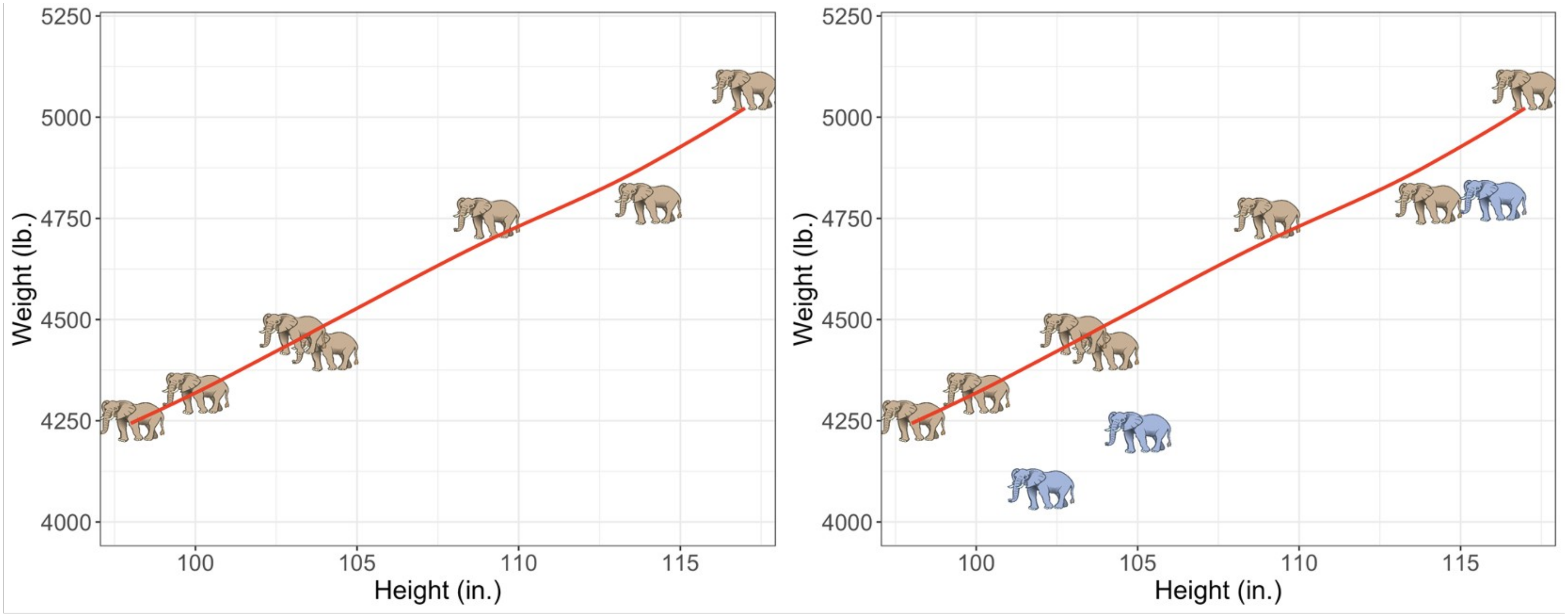
What Happens if We Only Run CV Once?

Training

Testing



Single Run CV Limitation



Our model's performance may be **sensitive to the sample** that we use for our training set.

Solution? Perform CV Multiple Times!

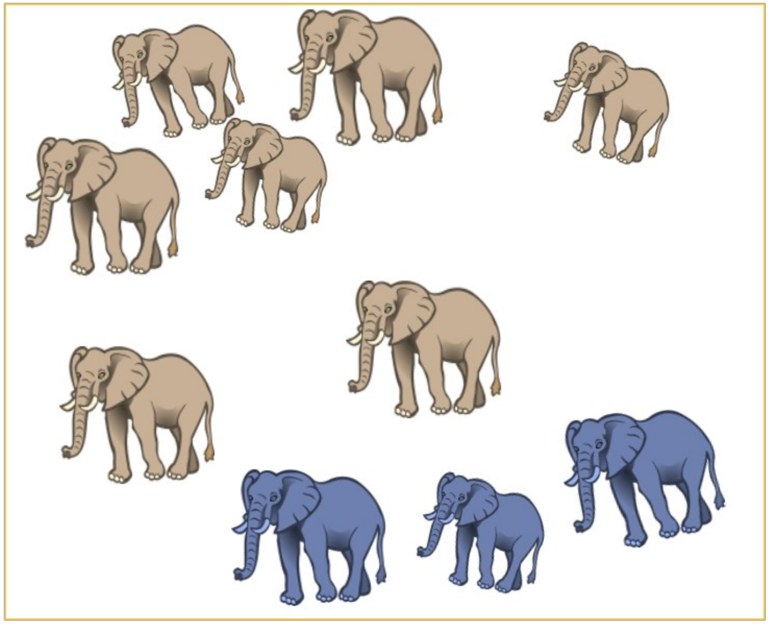
- To account for this issue, we can create many training and testing sets and assess the model's predictive ability multiple times.
- The results are then combined to get an overall estimate of a model's predictive performance.

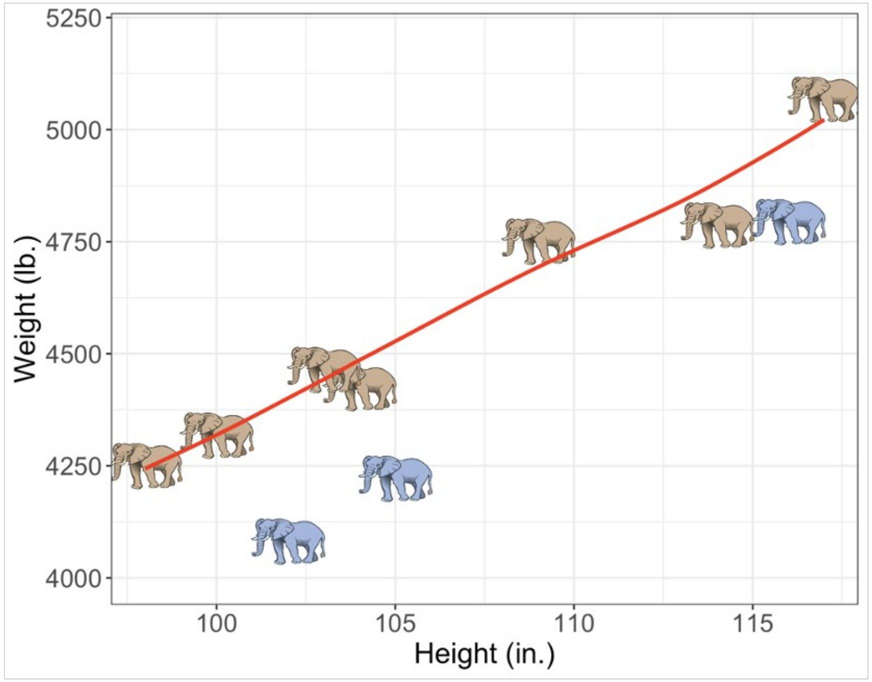
Performing CV multiple times gives a better representation of a model's performance.

CV Example Set 1

Training

Testing

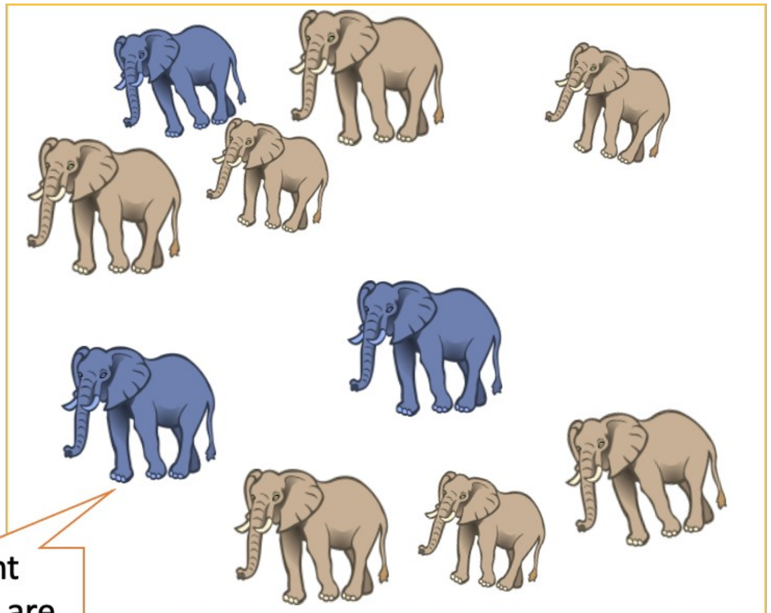




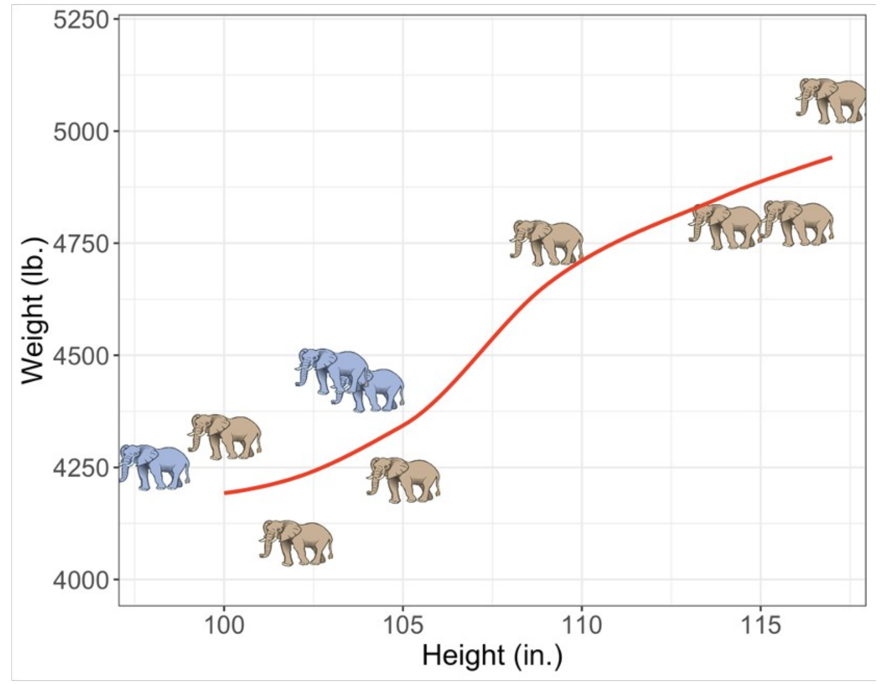
CV Example Set 2

Training

Testing



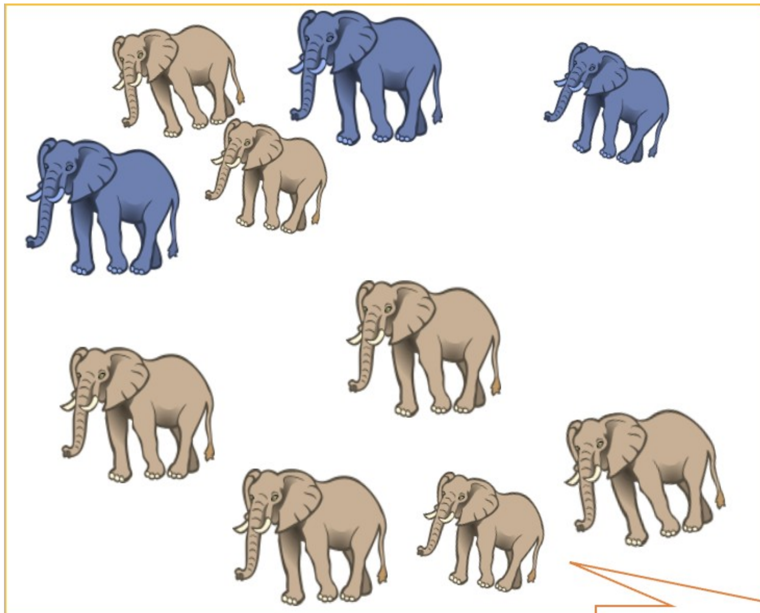
Different elephants are being used in the training & testing sets.

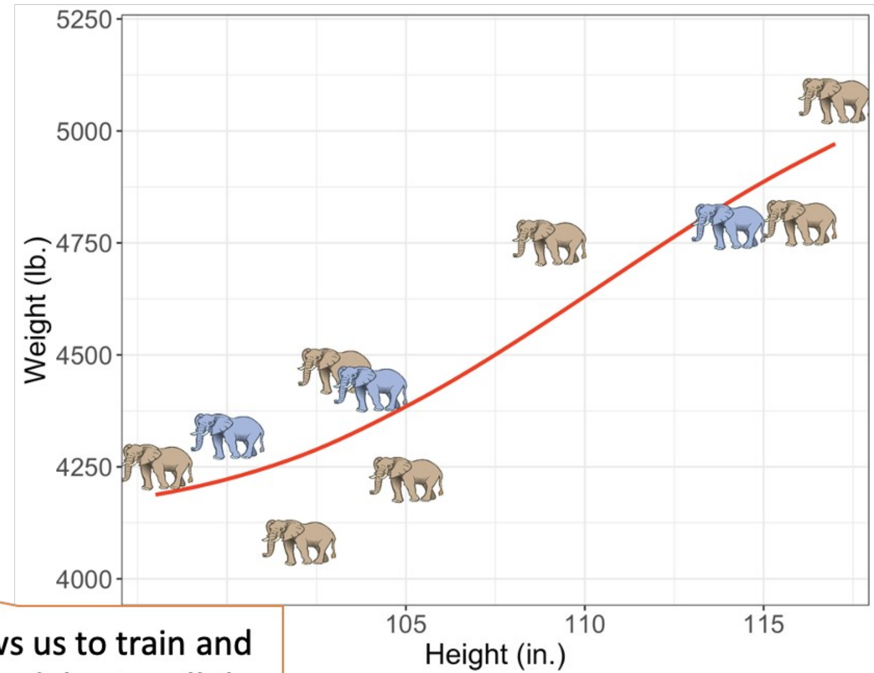


CV Example Set 3

Training

Testing



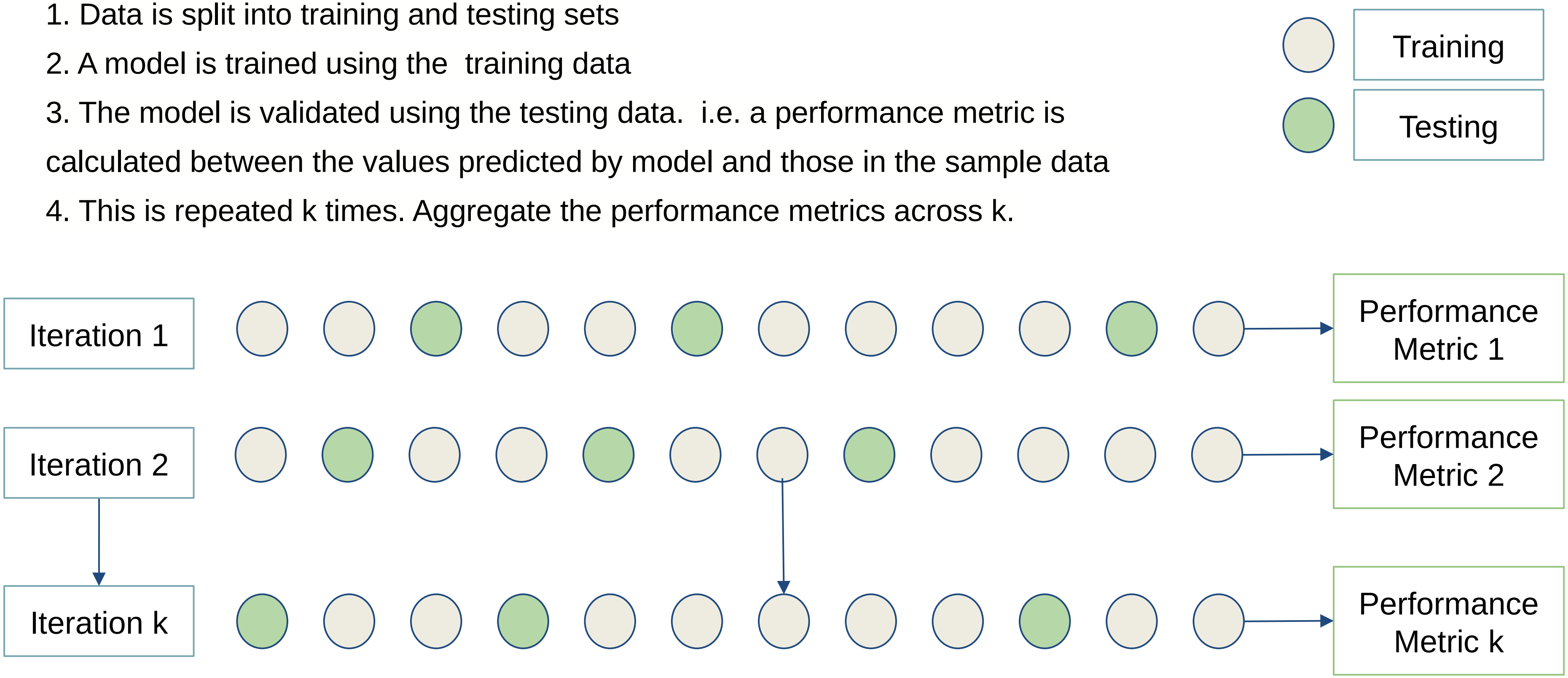


This allows us to train and test our model using all the different elephants we are interested in.

Cross Validation Overview

The cross validation process is generally split into the following steps:

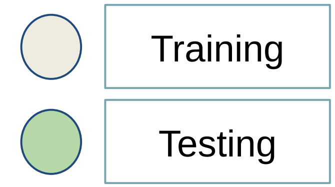
1. Data is split into training and testing sets
2. A model is trained using the training data
3. The model is validated using the testing data. i.e. a performance metric is calculated between the values predicted by model and those in the sample data
4. This is repeated k times. Aggregate the performance metrics across k.



$$overall\ performance = \frac{1}{k}\sum_{i=1}^{k} performance_i$$

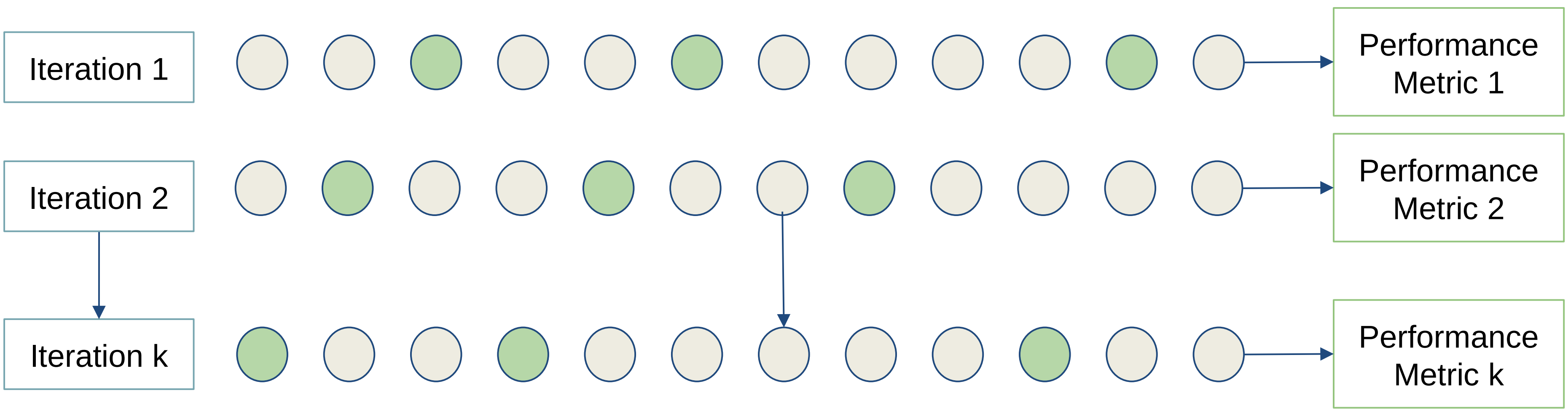
Given n, Choose k

$$\binom{n}{k} = \frac{n!}{k!(n-k)!}$$



Given n=12 samples, how many different ways can we choose k=9 samples?

12! / (9! * (12-9)!) = 12! / (9! * 3!) = 220



$$overall\ performance = \frac{1}{k}\sum_{i=1}^{k} performance_i$$

Error Metrics - Discrete Data*

Confusion Matrix

Can be used to calculate different error metrics that can be used to assess how well our model is performing.

		Actual	
		Positive	Negative
Predicted	Positive	True Positive	False Positive
	Negative	False Negative	True Negative

* Binary Data

Confusion Matrix Example

Testing Data Response	0	1	1	0	0	0	1	0	0	0	0
Predicted Response	0	0	1	0	0	0	0	1	0	0	0

		Actual	
		Positive (1)	Negative (0)
Predicted	Positive (1)		
	Negative (0)		

TP =
TN =
FP =
FN =

Confusion Matrix Example

Testing Data Response	0	1	1	0	0	0	1	0	0	0	0
Predicted Response	0	0	1	0	0	0	0	1	0	0	0

		Actual	
		Positive (1)	Negative (0)
Predicted	Positive (1)	1	1
	Negative (0)	2	7

TP = 1
TN = 7
FP = 1
FN = 2

Accuracy, Precision, Recall, F1

Accuracy

Accuracy = (TP+TN)/(TP+FP+FN+TN)
(no. of correct predictions/total no. of predictions)
When to use: Accuracy is a good choice when classes are balanced and not skewed.

Precision

Precision = (TP)/(TP+FP)
Of our predicted positives, what proportion is truly positive?
When to use: When we want to be very sure in our positive predictions.

Recall

Recall = (TP)/(TP+FN)
Of the actual positives, what proportion were accurately classified?
When to use: When we want to classify as many positives as possible.

F1

F1 = 2*(precision*recall)/(precision+recall)
Number in [0,1]. A harmonic mean of precision and recall.
When to use: When you want to have high precision and recall.

		Actual	
		Pos	Neg
Pred.	Pos	1 (TP)	1 (FP)
	Neg	2 (FN)	7 (TN)

Accuracy =
Precision =
Recall =
F1 =

Accuracy, Precision, Recall, F1

Accuracy

Accuracy = (TP+TN)/(TP+FP+FN+TN)
(no. of correct predictions/total no. of predictions)
When to use: Accuracy is a good choice when classes are balanced and not skewed.

Precision

Precision = (TP)/(TP+FP)
Of our predicted positives, what proportion is truly positive?
When to use: When we want to be very sure in our positive predictions.

Recall

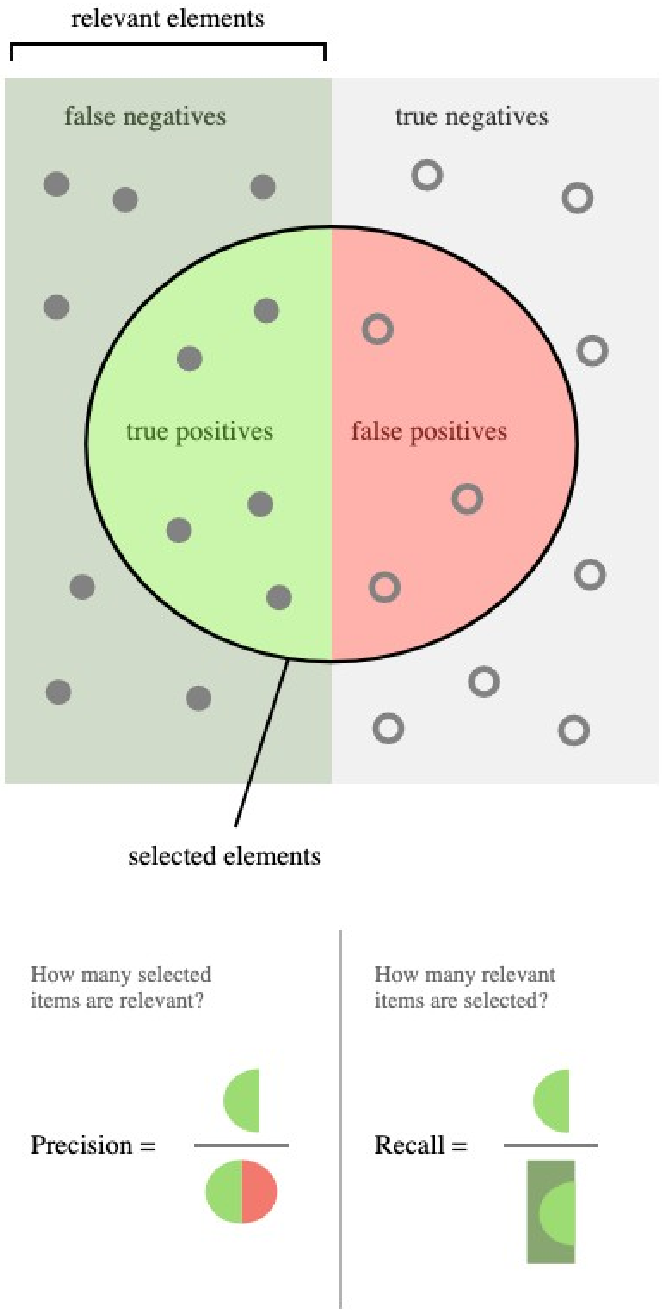
Recall = (TP)/(TP+FN)
Of the actual positives, what proportion were accurately classified?
When to use: When we want to classify as many positives as possible.

F1

F1 = 2*(precision*recall)/(precision+recall)
Number in [0,1]. A harmonic mean of precision and recall.
When to use: When you want to have high precision and recall.

		Actual	
		Pos	Neg
Pred.	Pos	1 (TP)	1 (FP)
	Neg	2 (FN)	7 (TN)

Accuracy = (1+7)/(1+1+2+7) = 8/11 = 73%
Precision = (1)/(1+1) = 1/2 = 50%
Recall = (1)/(1+2) = 1/3 = 33%
F1 = 2*(1/2)*(1/3)/(1/2+1/3) = 40%



source: wikipedia

[Many more metrics]