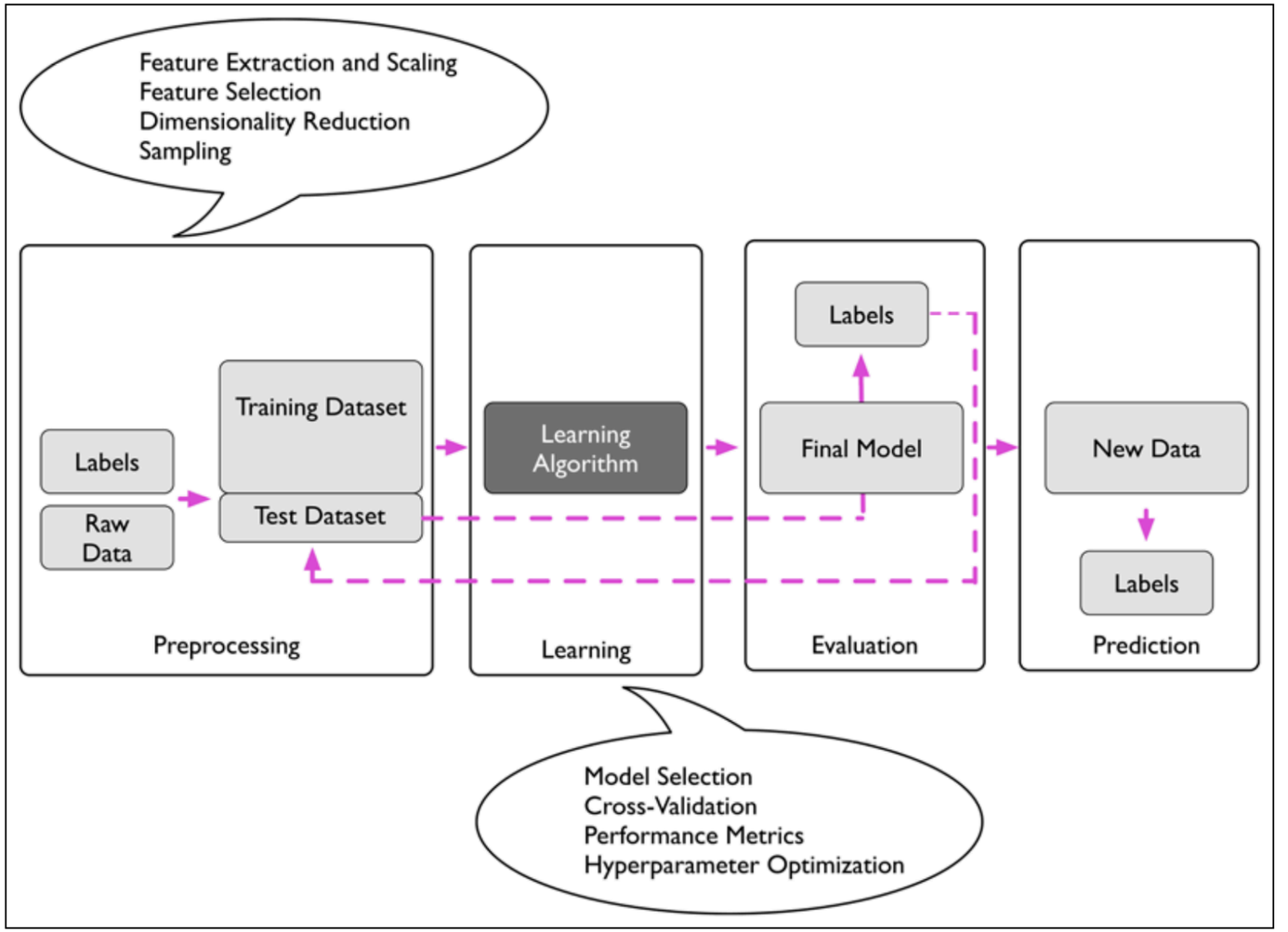
ML - Session 2
E2E ML

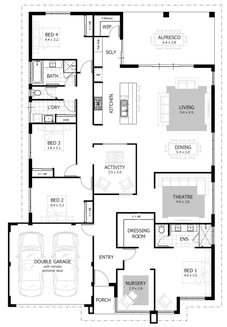
Previously on ML Sessions...



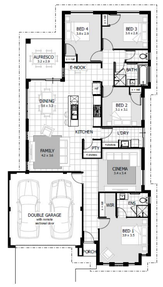
Example Problem

How to predict house price?

Let’s suppose that a construction company comes to us with a dataset of house plans images. They want to predict the house value using as an input the house plans.

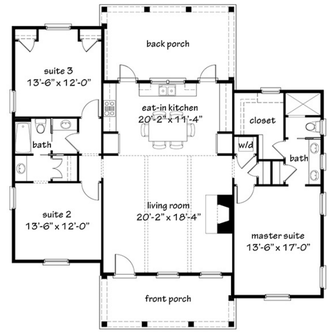


200.000 USD



300.000 USD

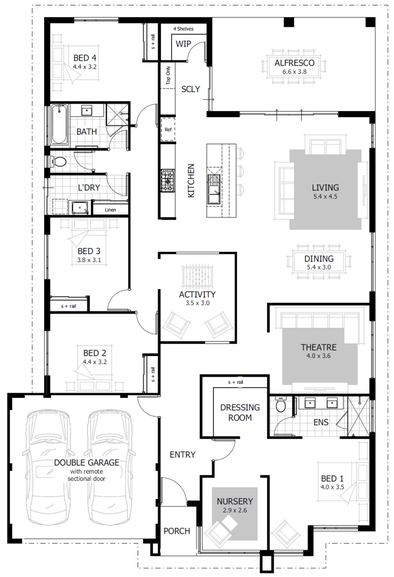
150.000 USD



500.000 USD

What do we have then?

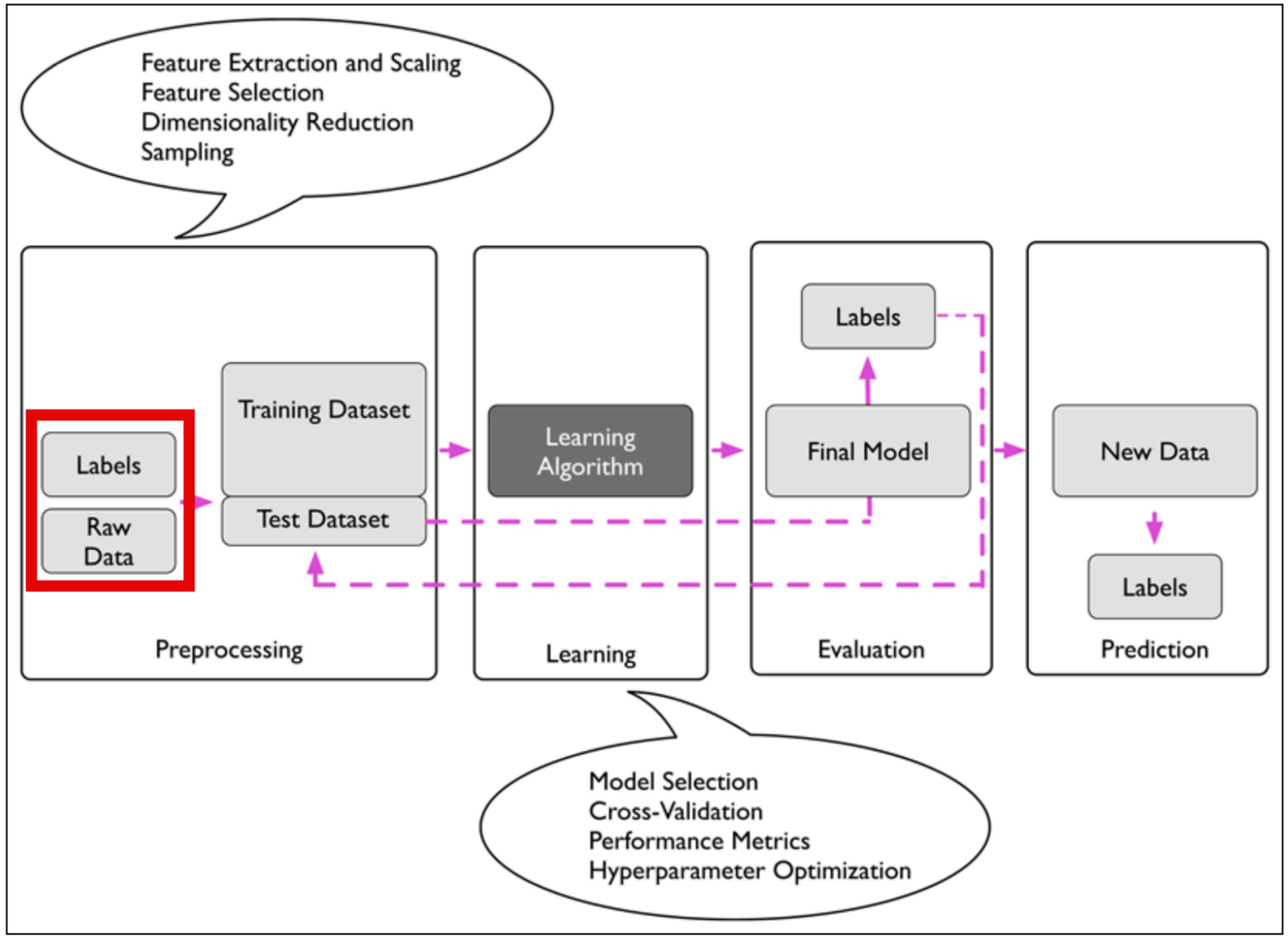
Raw data



Label

200.000 USD

Where are we?


Feature Extraction and Scaling
Feature Selection
Dimensionality Reduction
Sampling
Labels
Raw Data
Training Dataset
Test Dataset
Learning Algorithm
Labels
Final Model
New Data
Labels
Preprocessing
Learning
Evaluation
Prediction
Model Selection
Cross-Validation
Performance Metrics
Hyperparameter Optimization

Example Problem

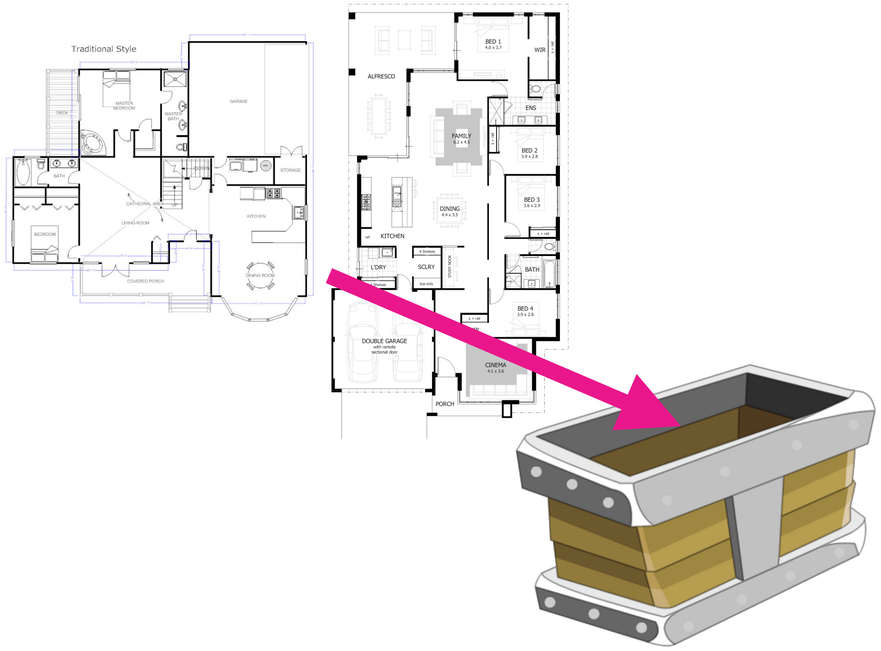
Data splitting

Training:

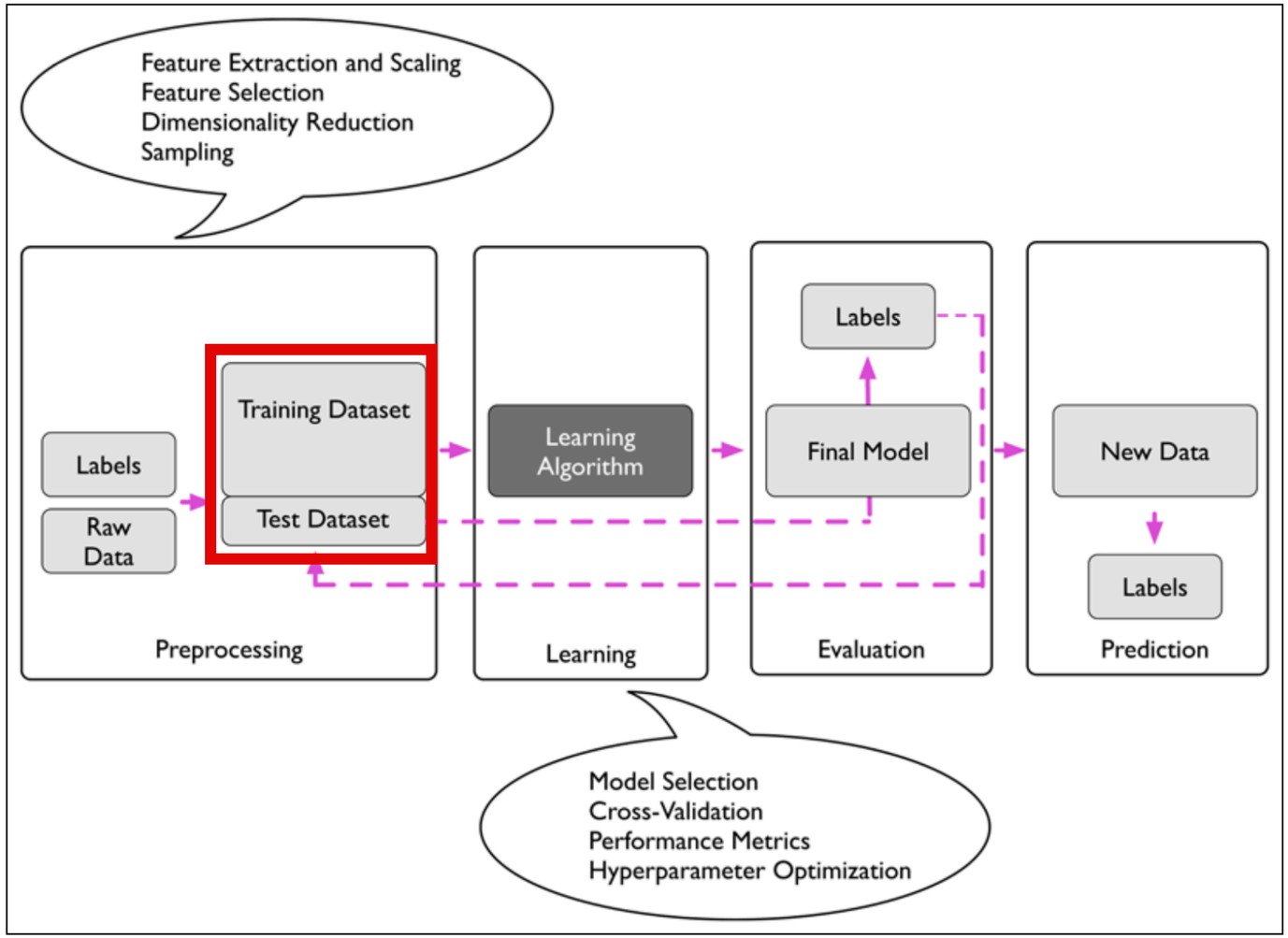
80% data

Test:

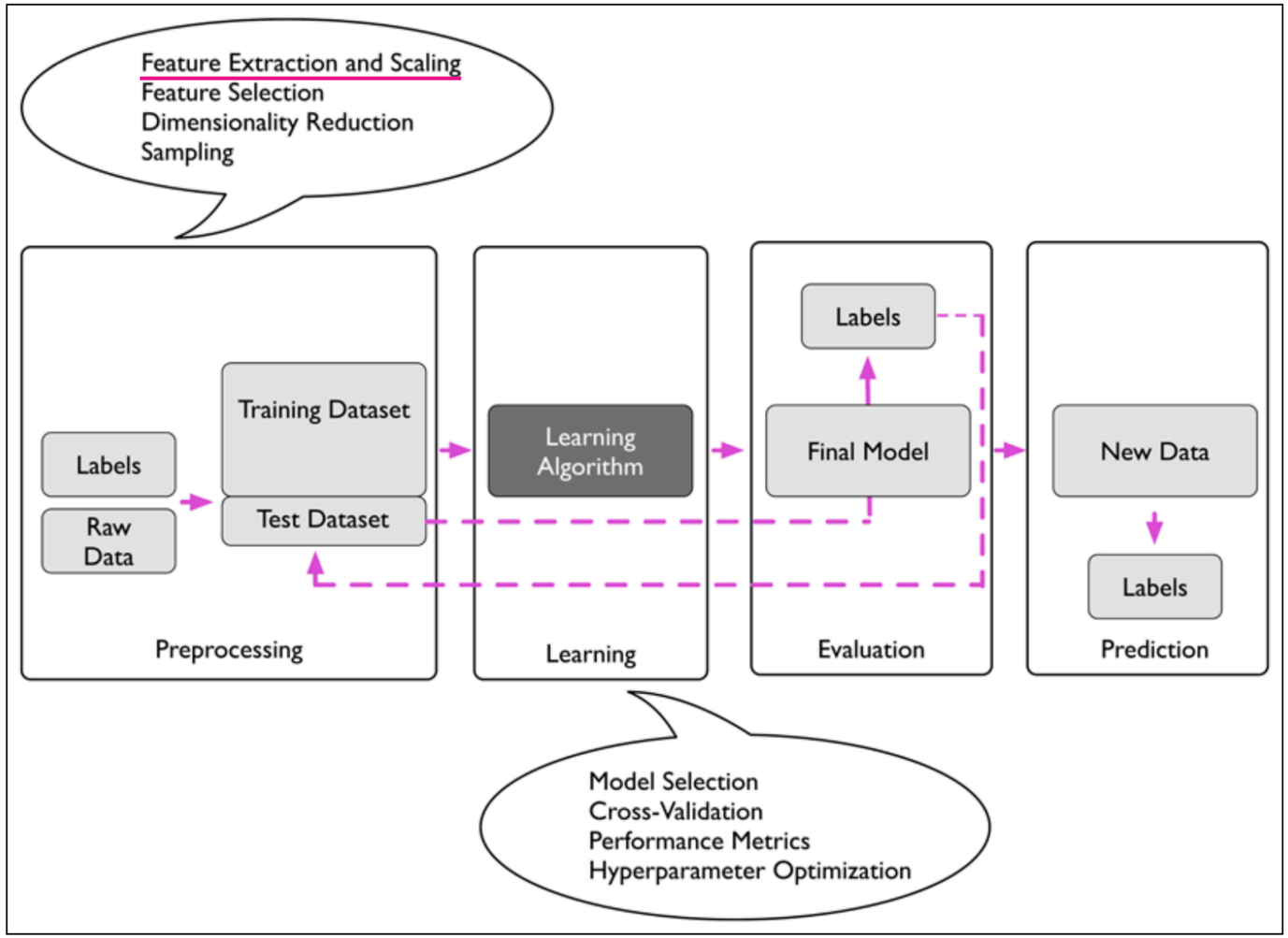
20% data



Where are we?

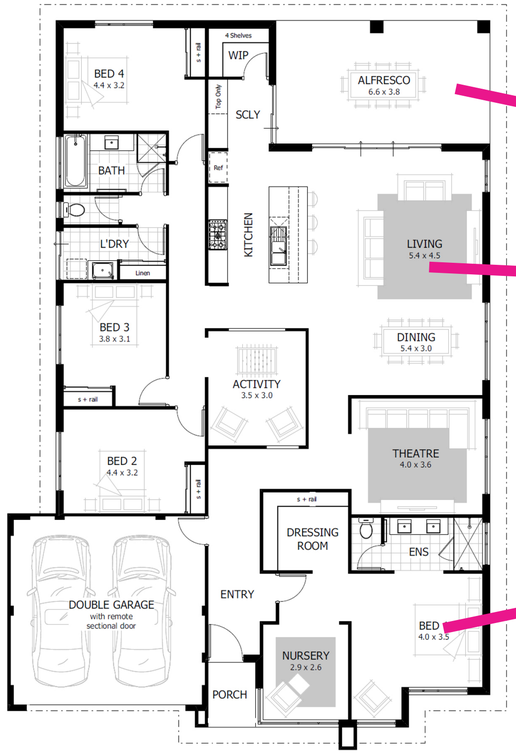

Feature Extraction and Scaling
Feature Selection
Dimensionality Reduction
Sampling
Labels
Raw Data
Training Dataset
Test Dataset
Learning Algorithm
Final Model
Labels
New Data
Labels
Preprocessing
Learning
Evaluation
Prediction
Model Selection
Cross-Validation
Performance Metrics
Hyperparameter Optimization

Where are we?


Feature Extraction and Scaling
Feature Selection
Dimensionality Reduction
Sampling
Labels
Raw Data
Training Dataset
Test Dataset
Learning Algorithm
Labels
Final Model
New Data
Labels
Preprocessing
Learning
Evaluation
Prediction
Model Selection
Cross-Validation
Performance Metrics
Hyperparameter Optimization

Example Problem

Feature extraction



computeBalconySize()

computeDinningSize()

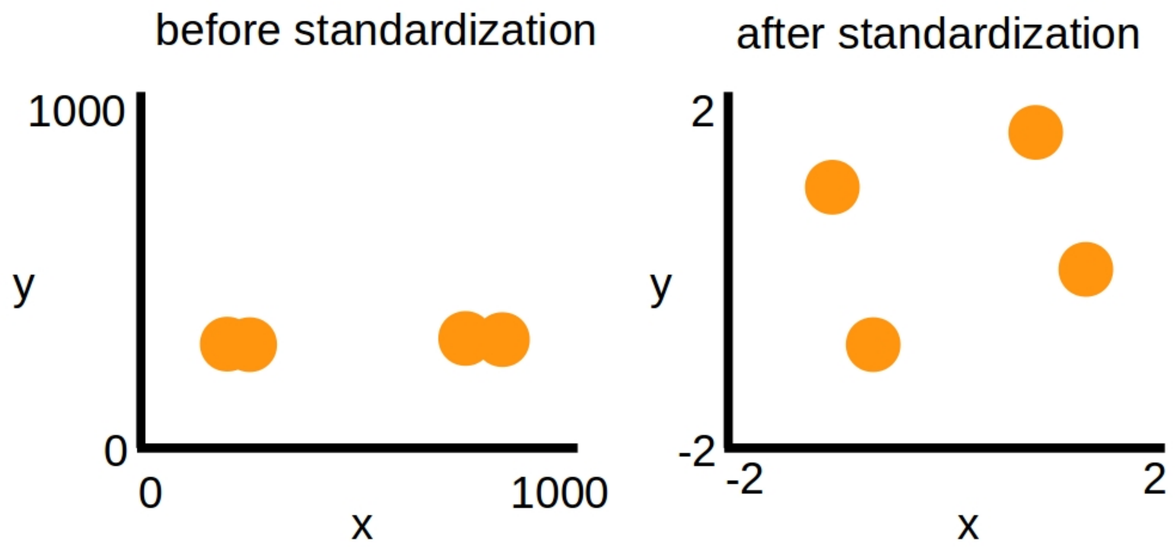
computeRoomySize()

$$X = \begin{bmatrix} 1 & x_{1,1} & x_{1,2} & \cdots & x_{1,m} \\ 1 & x_{2,1} & x_{2,2} & \cdots & x_{2,m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & x_{n,1} & x_{n,2} & \cdots & x_{n,m} \end{bmatrix}$$

200.000

200.000 USD

Feature scaling

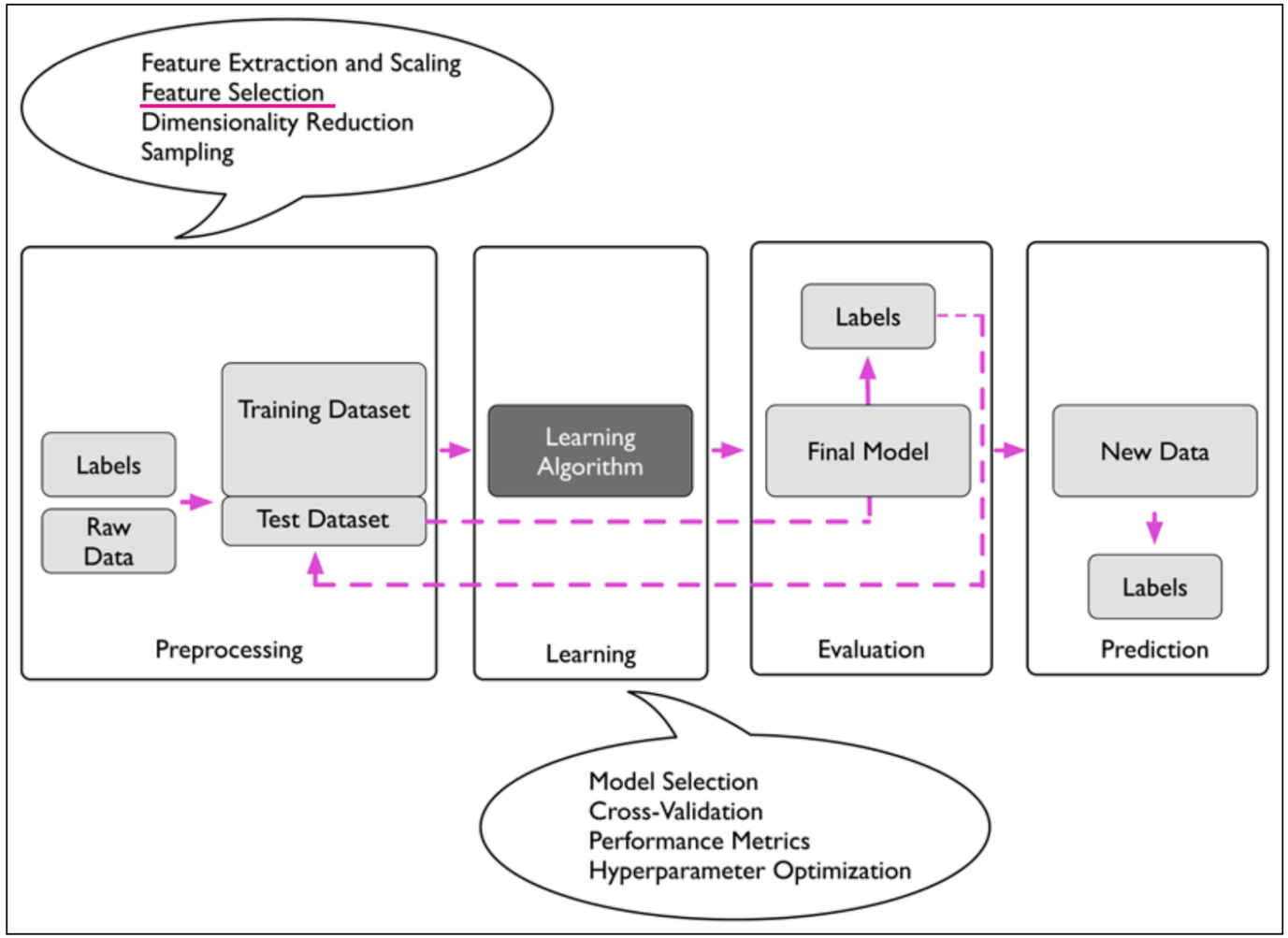


Example Problem

Feature scaling intuitive

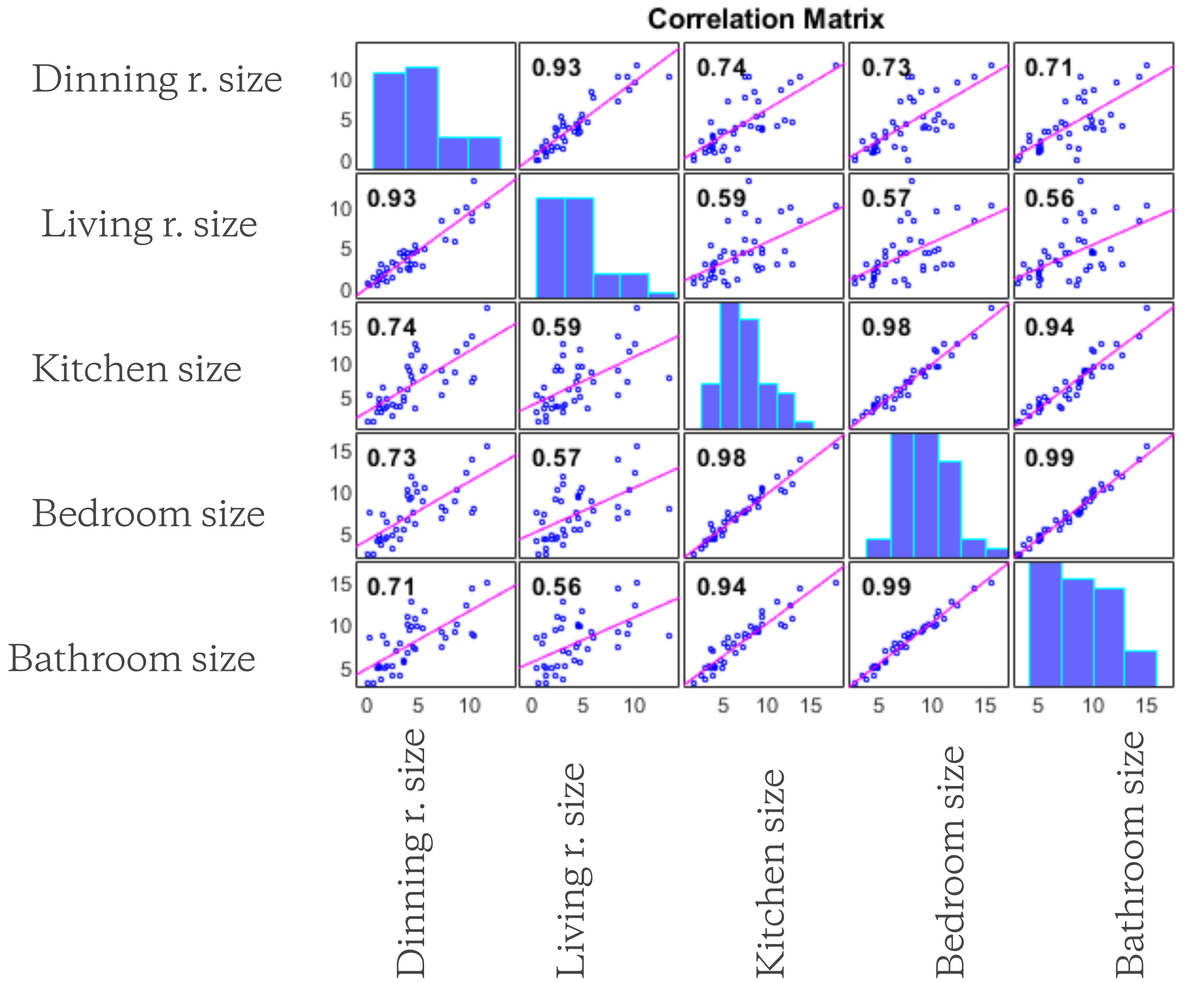
- A fisherman wants to be able to measure the similarity between two salmon, for example because he wants to classify them into two types for sale and thus be able to sell the most expensive large ones.
- For each salmon, measure its width and length.
- The length of fish salmon is a random variable that takes values between 50 and 100cm, while its width is between 10 and 20cm.
- However, as the differences in width are less substantial than those in length, it will be giving them less importance to the width.

Where are we?

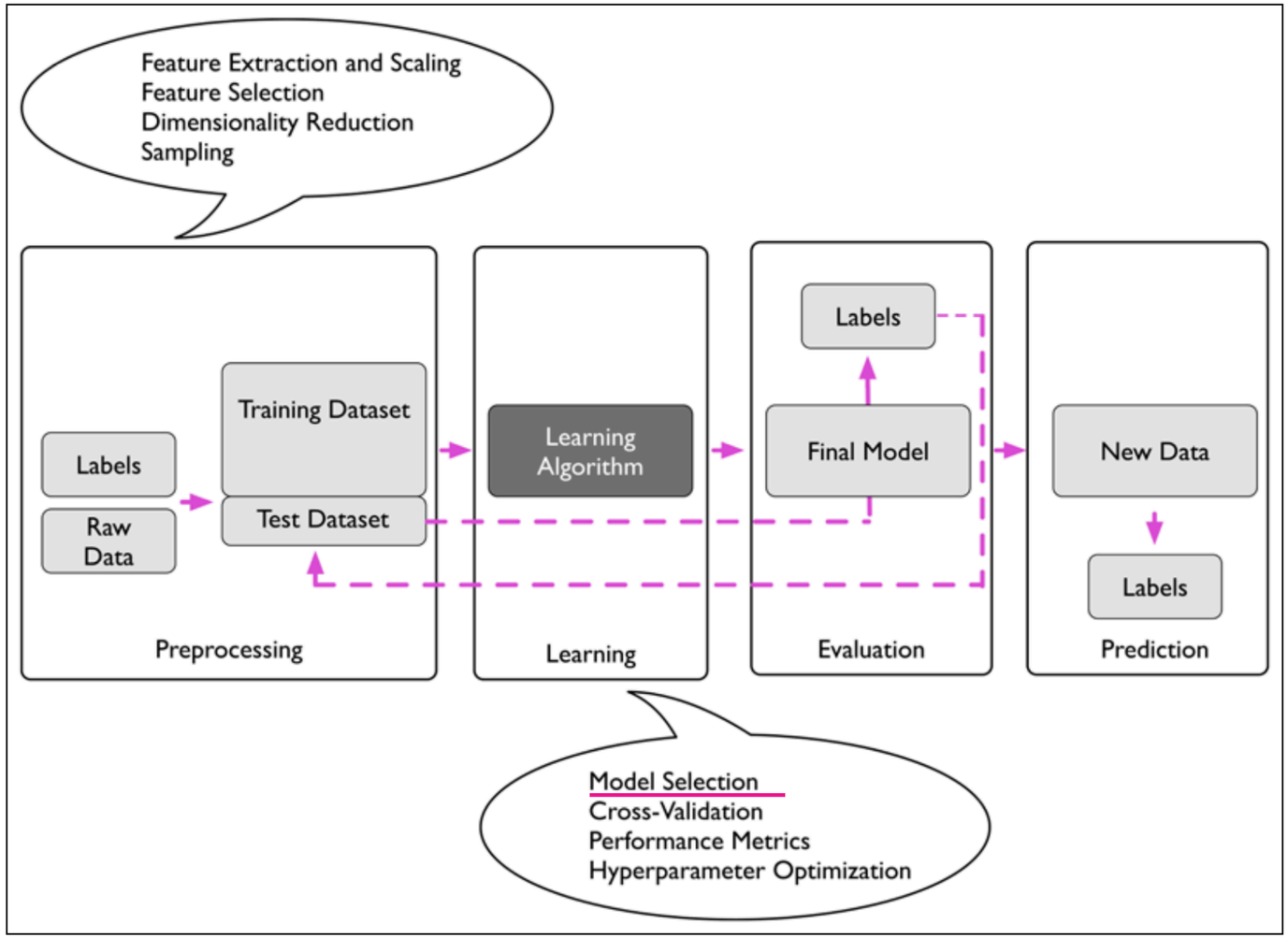

Feature Extraction and Scaling
Feature Selection
Dimensionality Reduction
Sampling
Labels
Raw Data
Training Dataset
Test Dataset
Preprocessing
Learning Algorithm
Learning
Labels
Final Model
Evaluation
New Data
Labels
Prediction
Model Selection
Cross-Validation
Performance Metrics
Hyperparameter Optimization

Example Problem

Feature selection

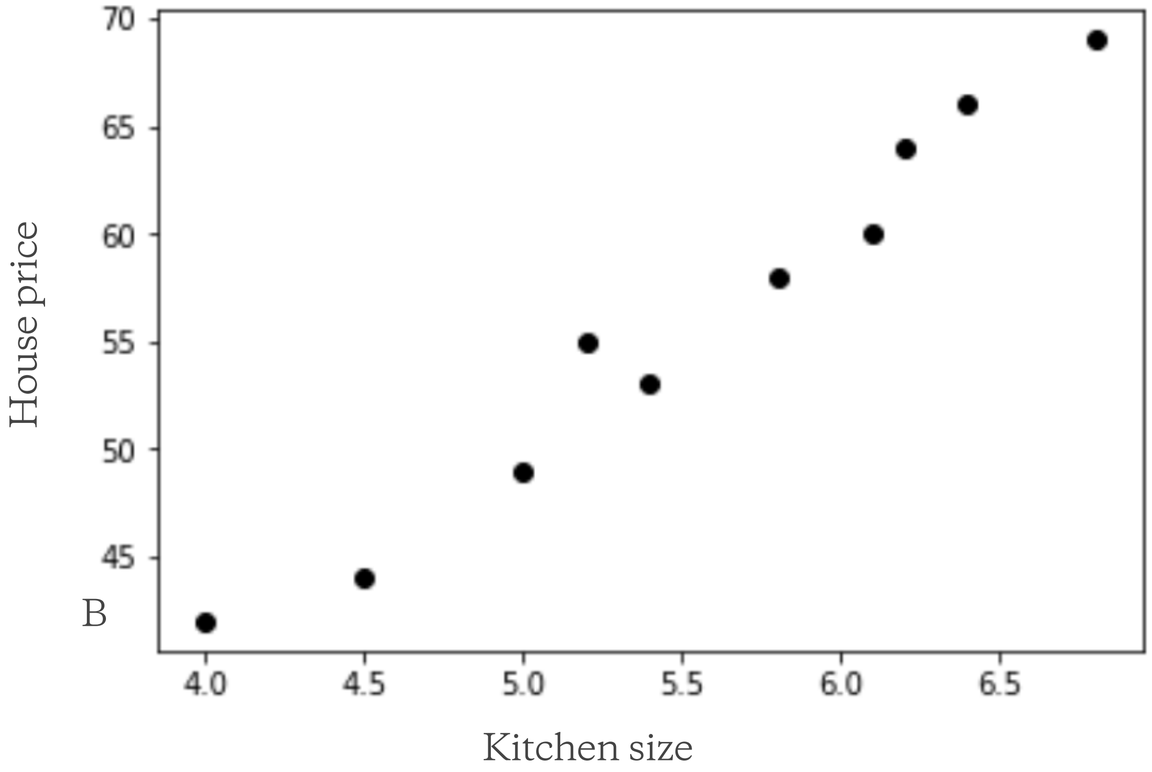


Where are we?

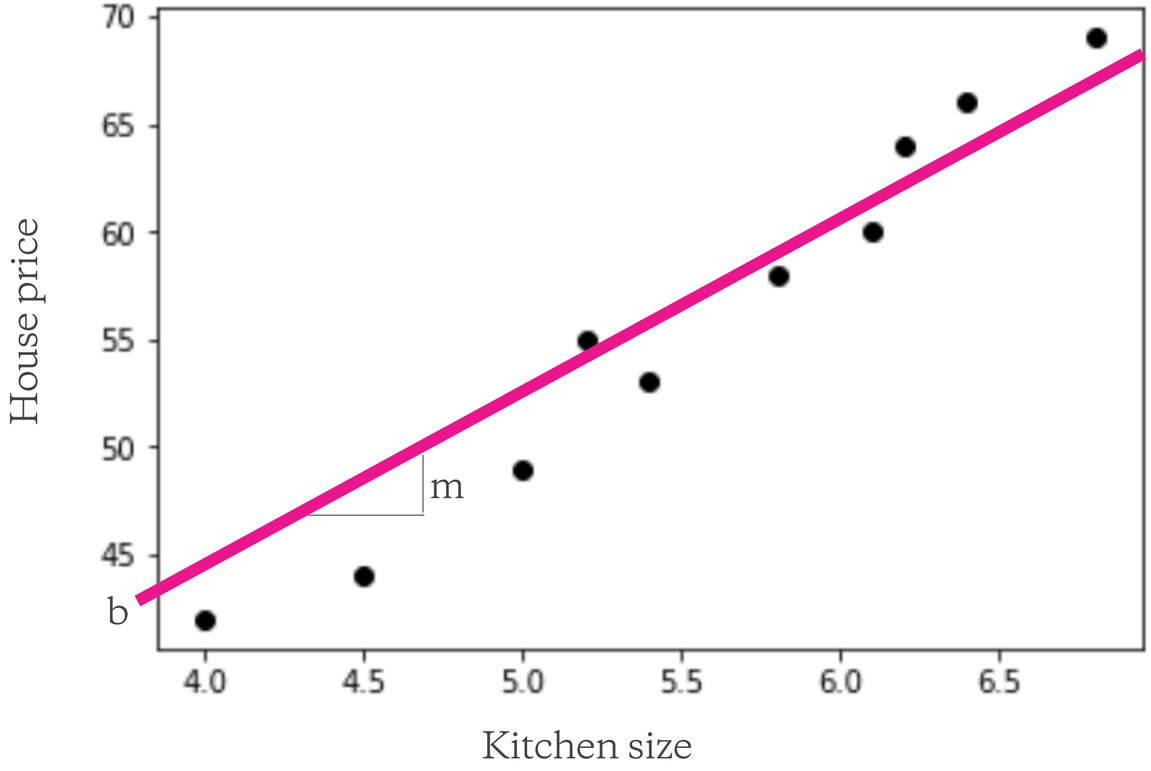

Feature Extraction and Scaling
Feature Selection
Dimensionality Reduction
Sampling
Labels
Raw Data
Training Dataset
Test Dataset
Learning Algorithm
Labels
Final Model
New Data
Labels
Preprocessing
Learning
Evaluation
Prediction
Model Selection
Cross-Validation
Performance Metrics
Hyperparameter Optimization

Example Problem

Model



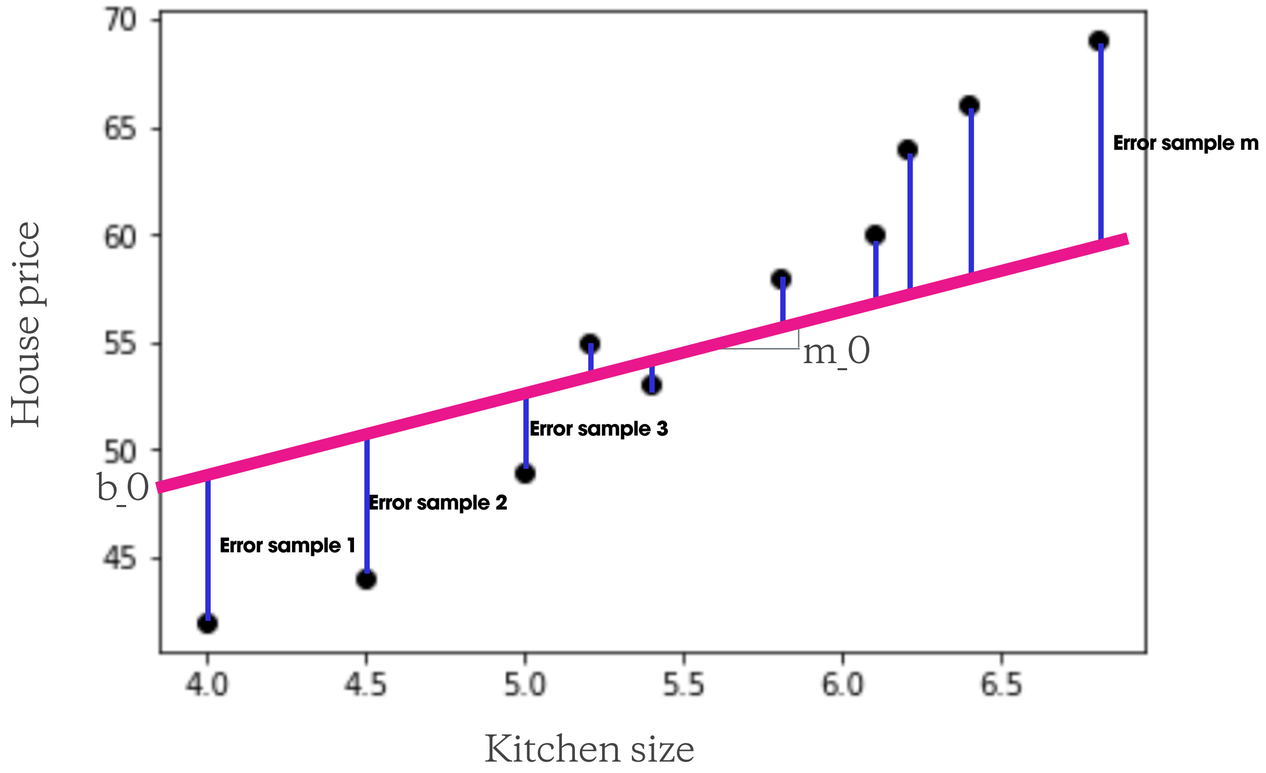
y = m*x + b



Output = m * kitchenSize + b

Example Problem

Learning Algorithm



$$MSE = \frac{1}{n}\Sigma\underbrace{\left(y - \hat{y}\right)}^{2}$$

The square of the difference between actual and predicted

An algorithm will run p iterations trying to minimize the MSE

Output = m_0 * kitchenSize + b_0

Example Problem

Learning Algorithm

Parameters:

m and b

Hyperparameters:

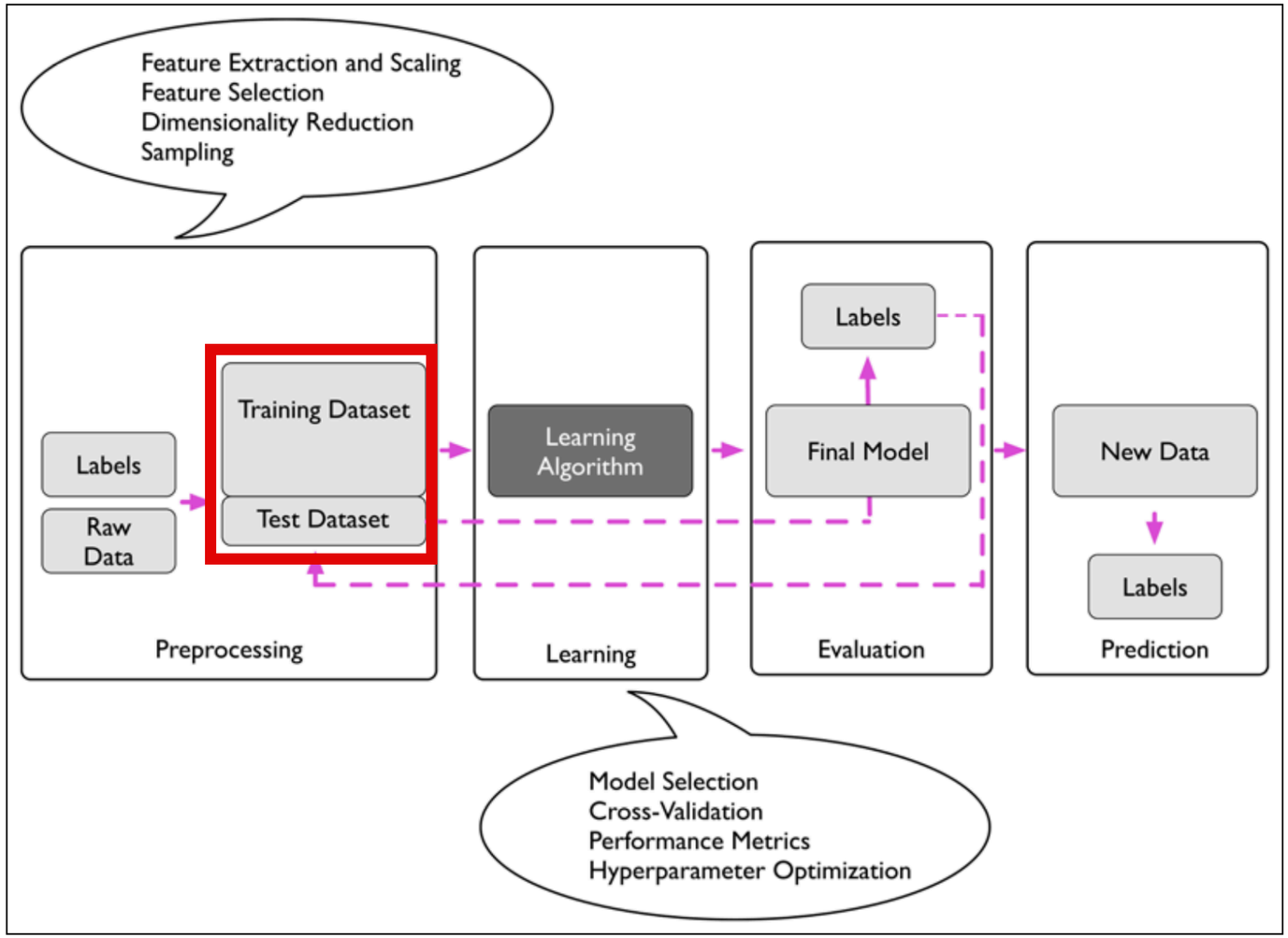
number of iteration p and learning rate

- How many iteration we must run the algorithm?
- How to pick the learning rate?

How do you choose the value of one hyperparameter?

- One option is to use a test set, train 100 different models using 100 different values for this hyperparameter.
- But, when you find the best model and launch it into production, but unfortunately it does not perform as well as expected. What just happened?

Where are we?

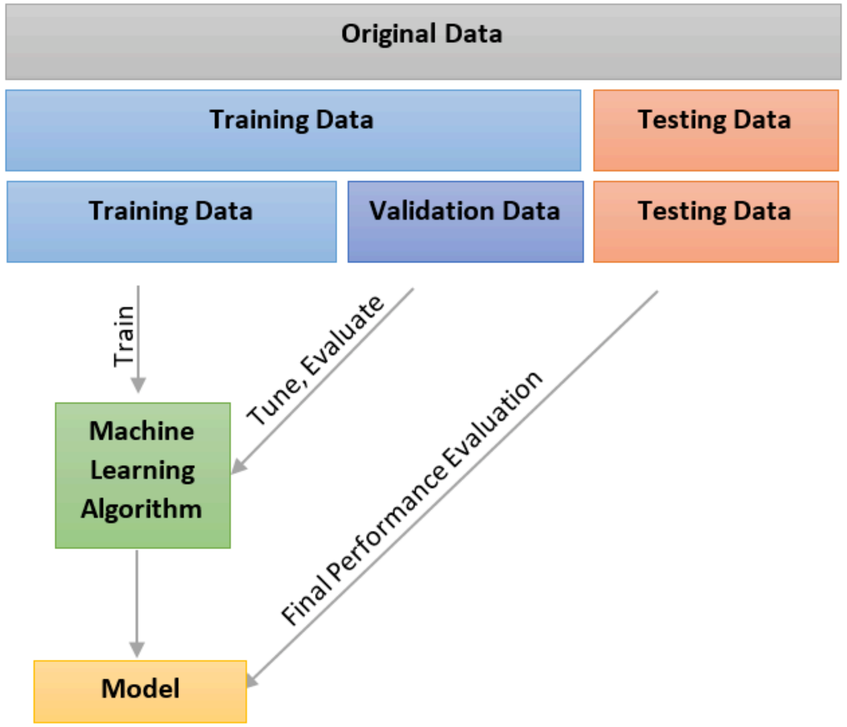

Feature Extraction and Scaling
Feature Selection
Dimensionality Reduction
Sampling
Labels
Raw Data
Training Dataset
Test Dataset
Learning Algorithm
Final Model
Labels
New Data
Labels
Preprocessing
Learning
Evaluation
Prediction
Model Selection
Cross-Validation
Performance Metrics
Hyperparameter Optimization

What happened?

- The generalization error was measured multiple times on the test set, and you adapted the model and hyperparameters to produce the best model for that particular set.
- This means that the model is unlikely to perform as well on new data.

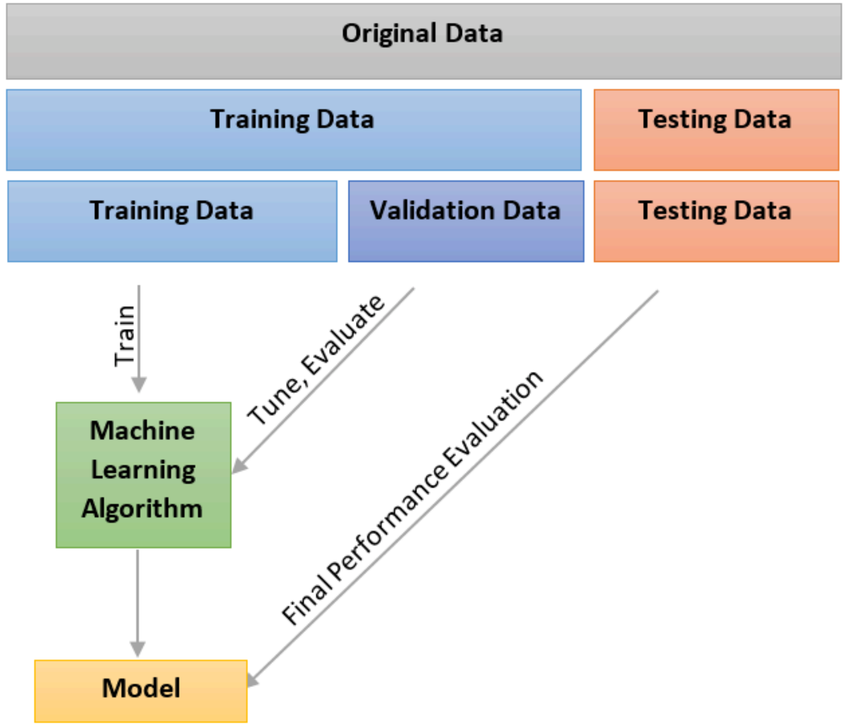
One solution: Validation

- Hold out part of the training set to evaluate several candidate models and select the best one. The new set is called the validation set (or sometimes the development set, or dev set).
- Train multiple models with various hyperparameters on the reduced training set (i.e., the full training set minus the validation set).



Hyperparameter Tuning

- Select the model that performs best on the validation set.
- After this, you train the best model on the full training set (including the valida- tion set), and this gives you the final model.
- Lastly, you evaluate this final model on the test set to get an estimate of the generalization error.

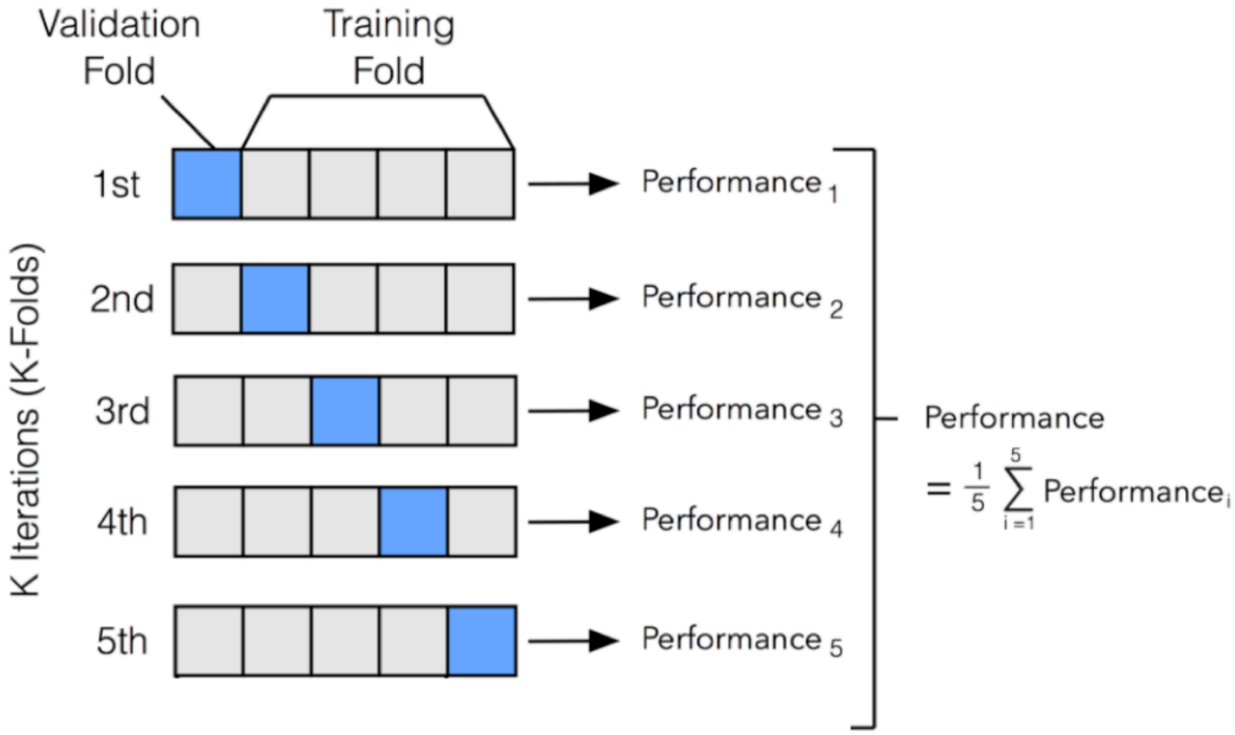


What problems can have this solution?

- If validation set is too small, model evaluations will be imprecise and model selection may be suboptimal.
- If the validation set is too large, then the remaining training set will be much smaller than the full training set. The final model will be trained on the full training set and it is not ideal to compare candidate models trained on a much smaller training set.

Cross-validation

- Using many small validation sets.
- By averaging out all the evaluations of a model, we get a much more accurate measure of its performance. Drawback: the training time is multiplied by the number of validation sets.



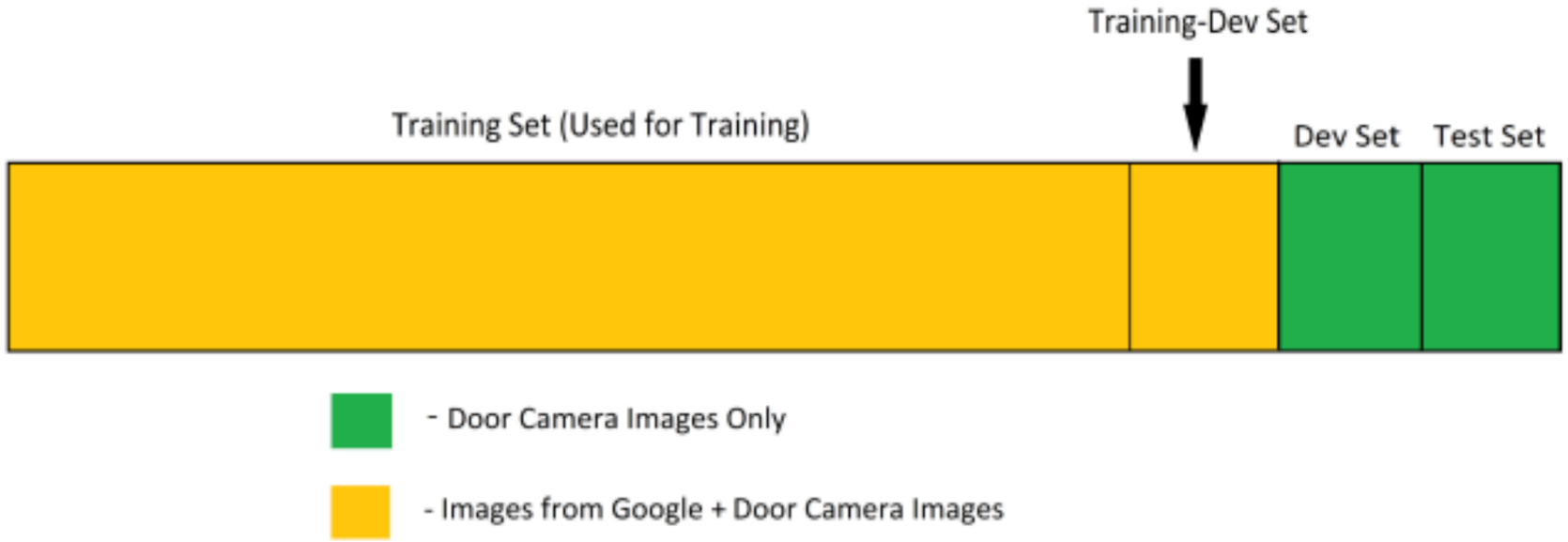
Data Mismatch

- In some cases, it is easy to get a large amount of data for training, but it is not perfectly representative of the data that will be used in production.
- In this case, the most important rule to remember is that the validation set and the test must be as representative as possible of the data you expect to use in production, so they should be composed exclusively of representative pictures.

Dataset

Production data

When is an overfitting problem and when is data mismatch?



Data Mismatch

- Since the training-dev set has the same distribution as the training set and the training-dev error is higher, it is safe to assume that we have a variance problem (overfitting).

Training Error – 1%

Training-Dev Error – 9%

Dev Error – 10%

Data Mismatch

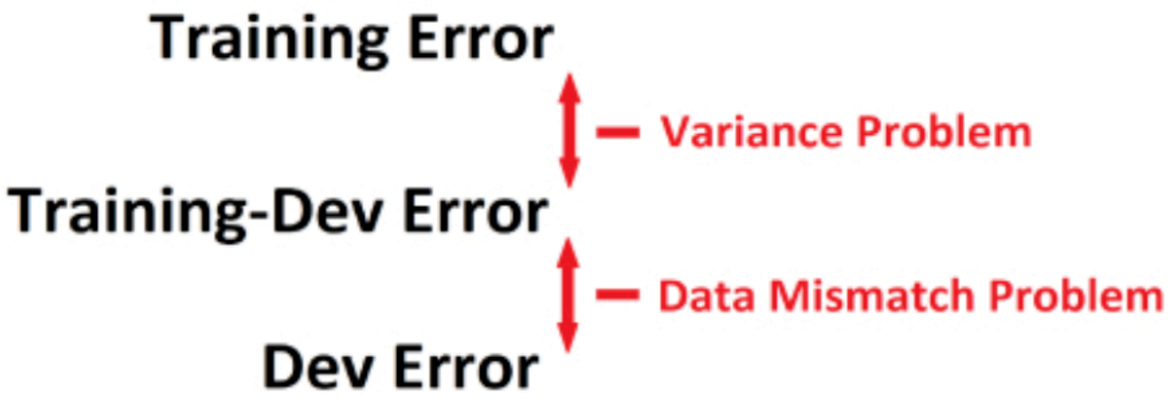
- The training error and the training-dev error are very close but there is a high dev error. This shows that our model is suffering from a data mismatch problem.

Training Error – 1%

Training-Dev Error – 1.5%

Dev Error – 10%

Data Mismatch



No Free Lunch Theorem

- A model is a simplified version of the observations. The simplifications are meant to discard the superfluous details that are unlikely to generalize to new instances.
- However, to decide what data to discard and what data to keep, you must make assumptions.
- In a famous 1996 paper, 11 David Wolpert demonstrated that if you make absolutely no assumption about the data, then there is no reason to prefer one model over any other. This is called the No Free Lunch (NFL) theorem.

No Free Lunch Theorem

- The only way to know for sure which model is best is to evaluate them all.

- Since this is not possible, in practice you make some reasonable assumptions about the data and you evaluate only a few reasonable models.

Main steps you will go through:

- Look at the big picture.
- Get the data.
- Discover and visualize the data to gain insights.
- Prepare the data for Machine Learning algorithms.
- Select a model and train it.
- Fine-tune your model.
- Present your solution.
- Launch, monitor, and maintain your system