SUPERVISED LEARNING REGRESSION

WHAT WILL WE COVER?

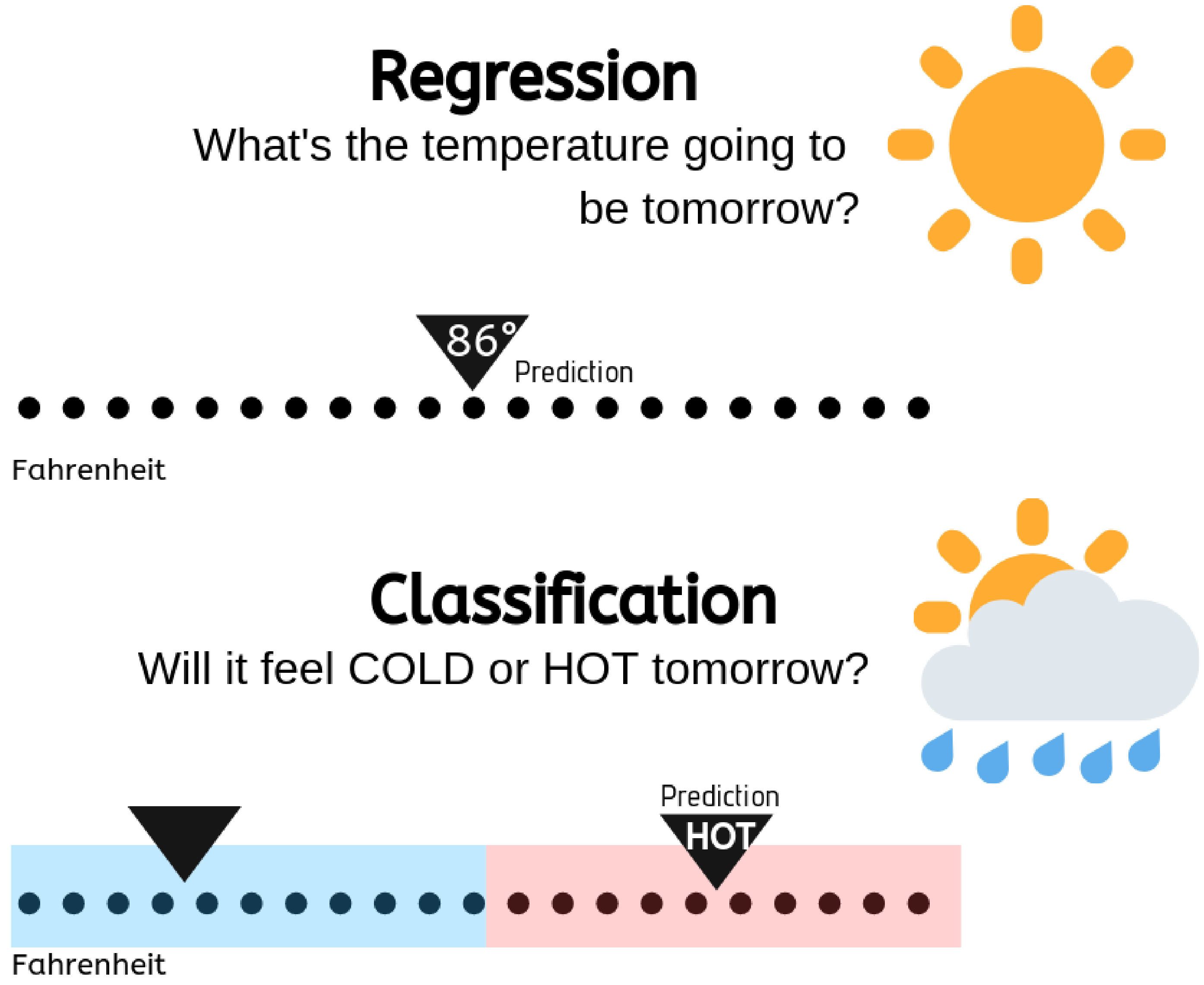
- Regression Algorithms
- Regression Algorithms Evaluation

MACHINE LEARNING: TYPES OF SUPERVISED PROBLEMS

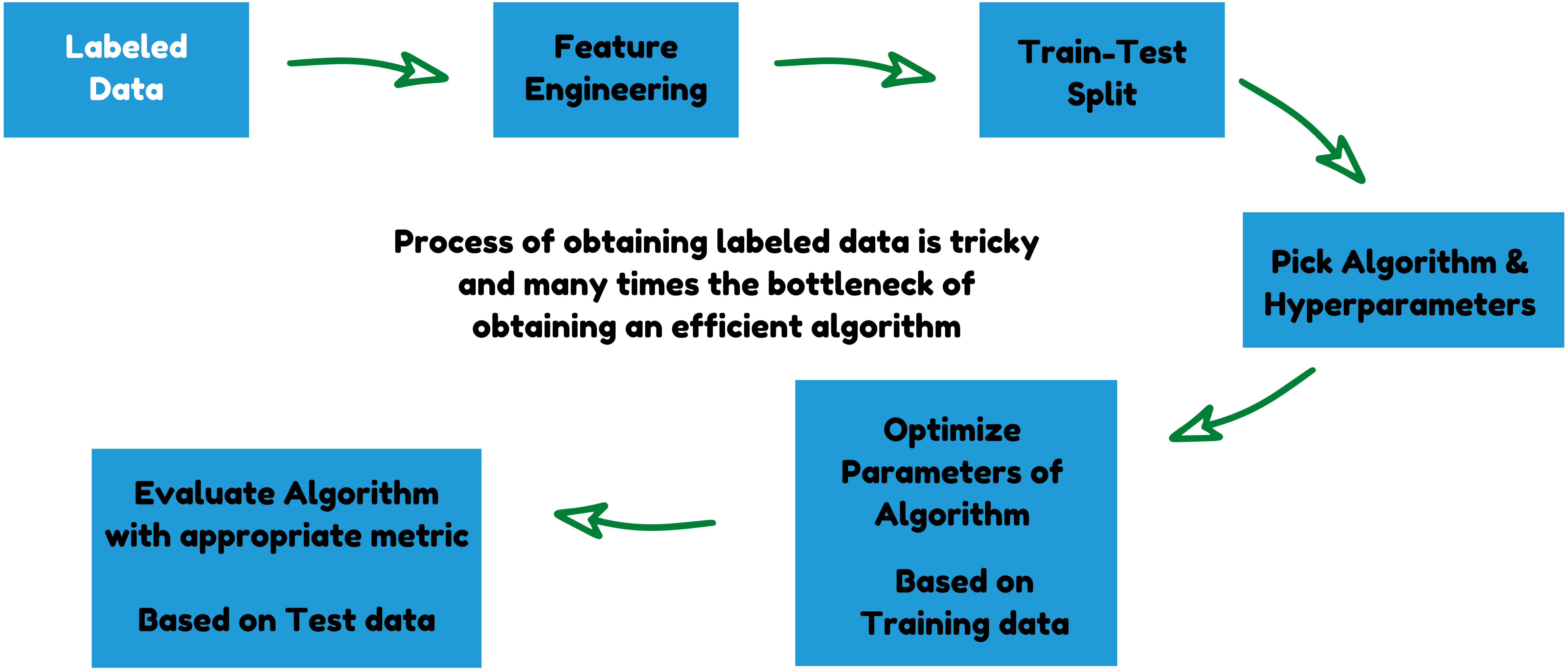
- **In Machine Learning a particular problem can either be a Classification or Regression problem.**
- **The difference is in the type of variable we are trying to predict**

Continuous/Numerical : Regression Problem

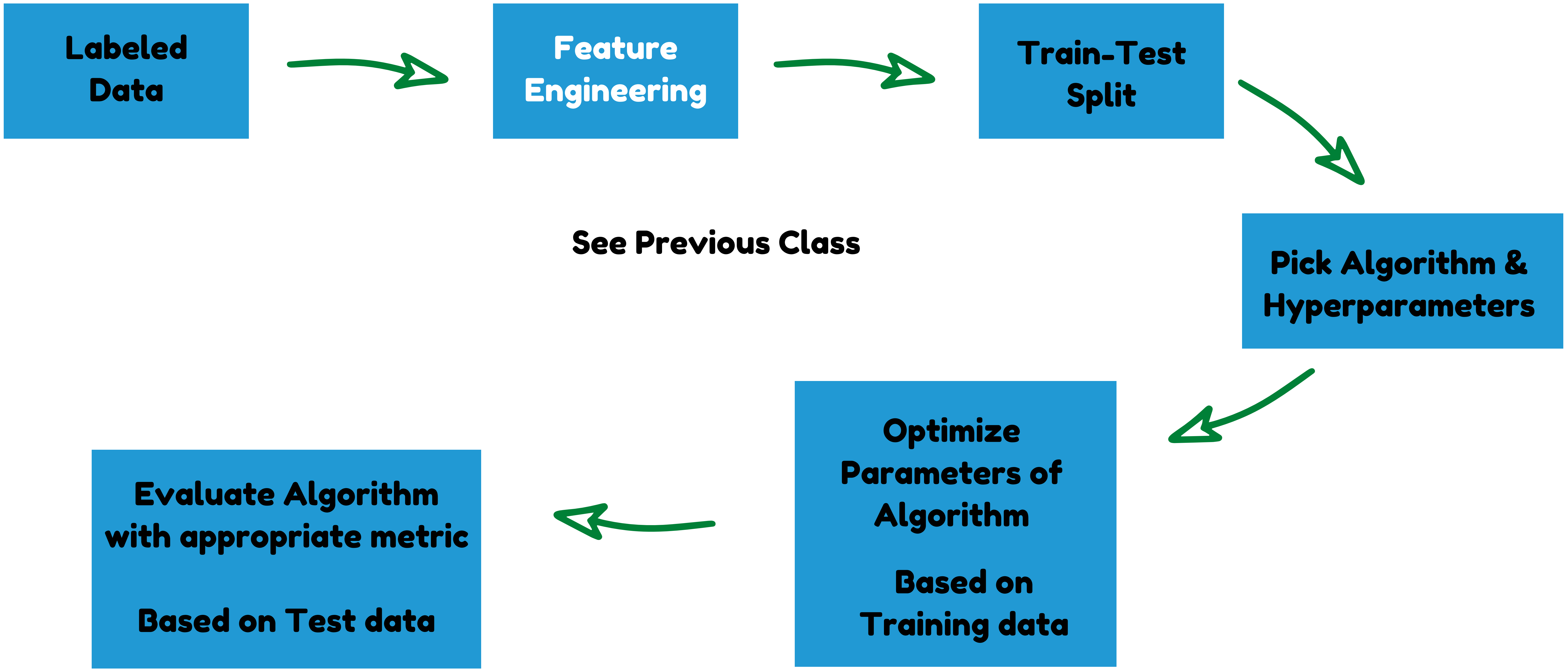
Categorical: Classification Problem



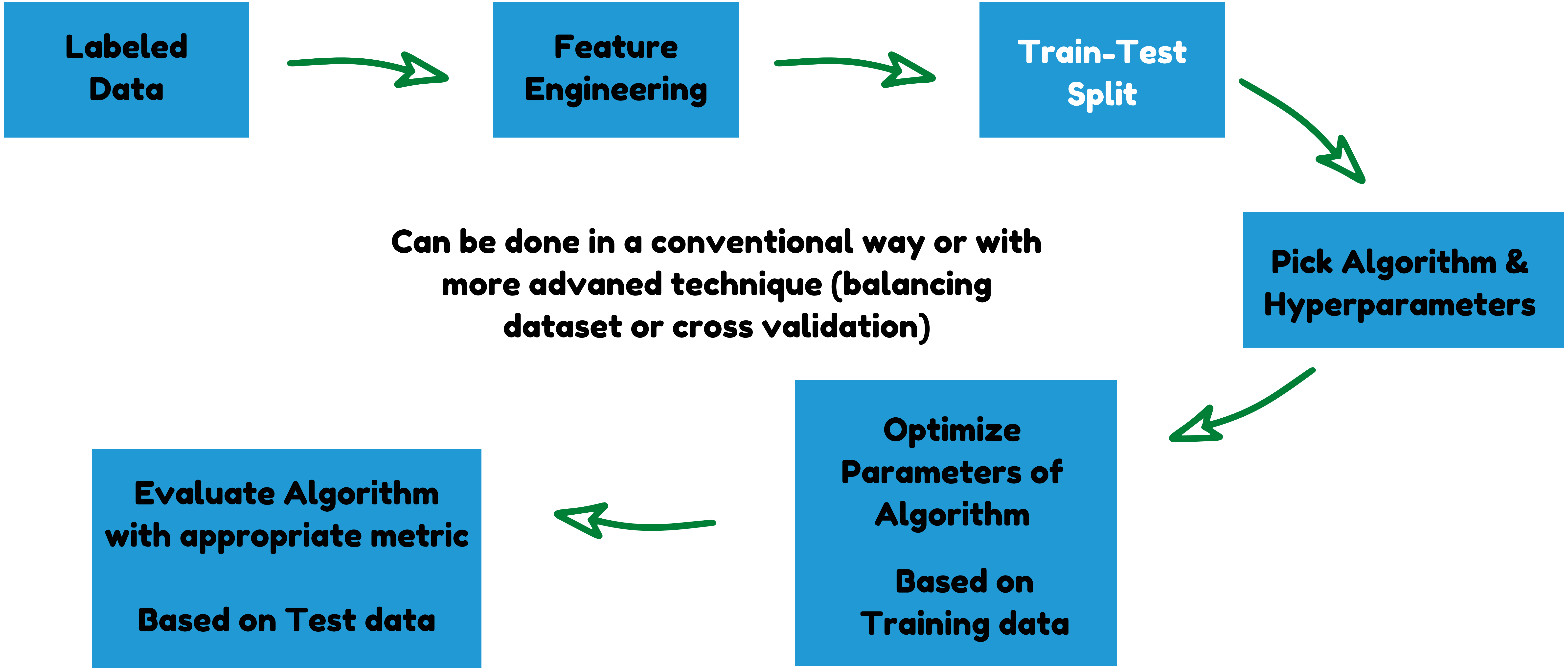
THE MACHINE LEARNING PROCESS



THE MACHINE LEARNING PROCESS



THE MACHINE LEARNING PROCESS



THE MACHINE LEARNING PROCESS

Labeled Data

Feature Engineering

Train-Test Split

Pick Algorithm & Hyperparameters

This is where you earn your salary. With experience you will have intuition of which algorithm will be best suited for your situation. The hyperparameters have to be set

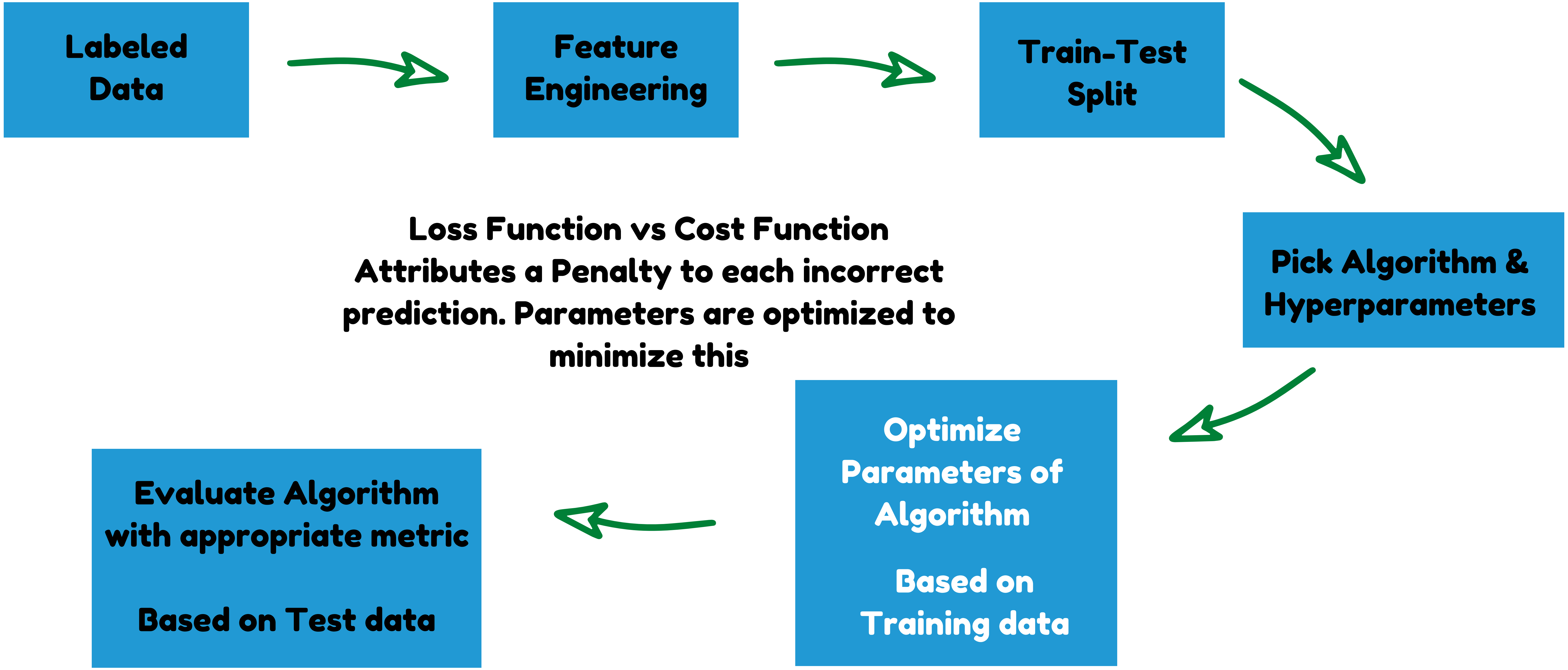
Optimize Parameters of Algorithm

Based on Training data

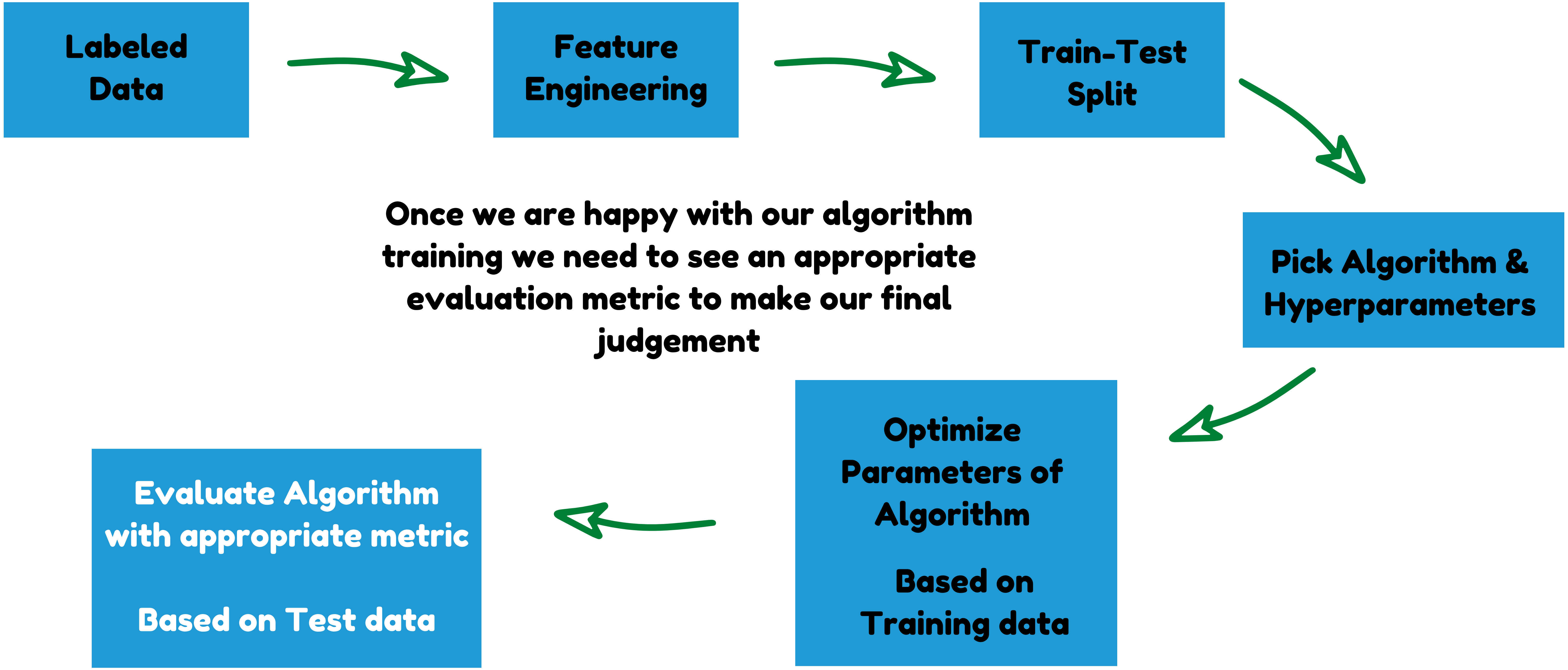
Evaluate Algorithm with appropriate metric

Based on Test data

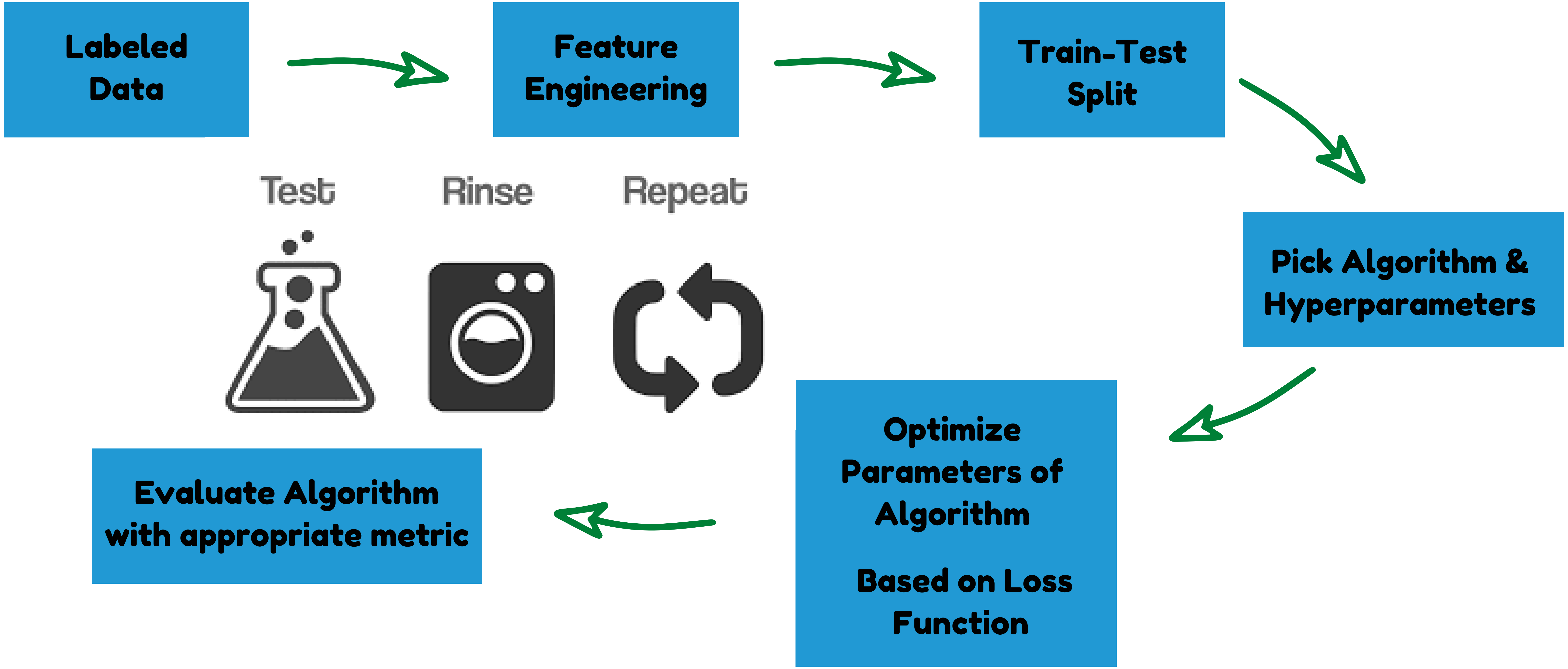
THE MACHINE LEARNING PROCESS



THE MACHINE LEARNING PROCESS



THE MACHINE LEARNING PROCESS

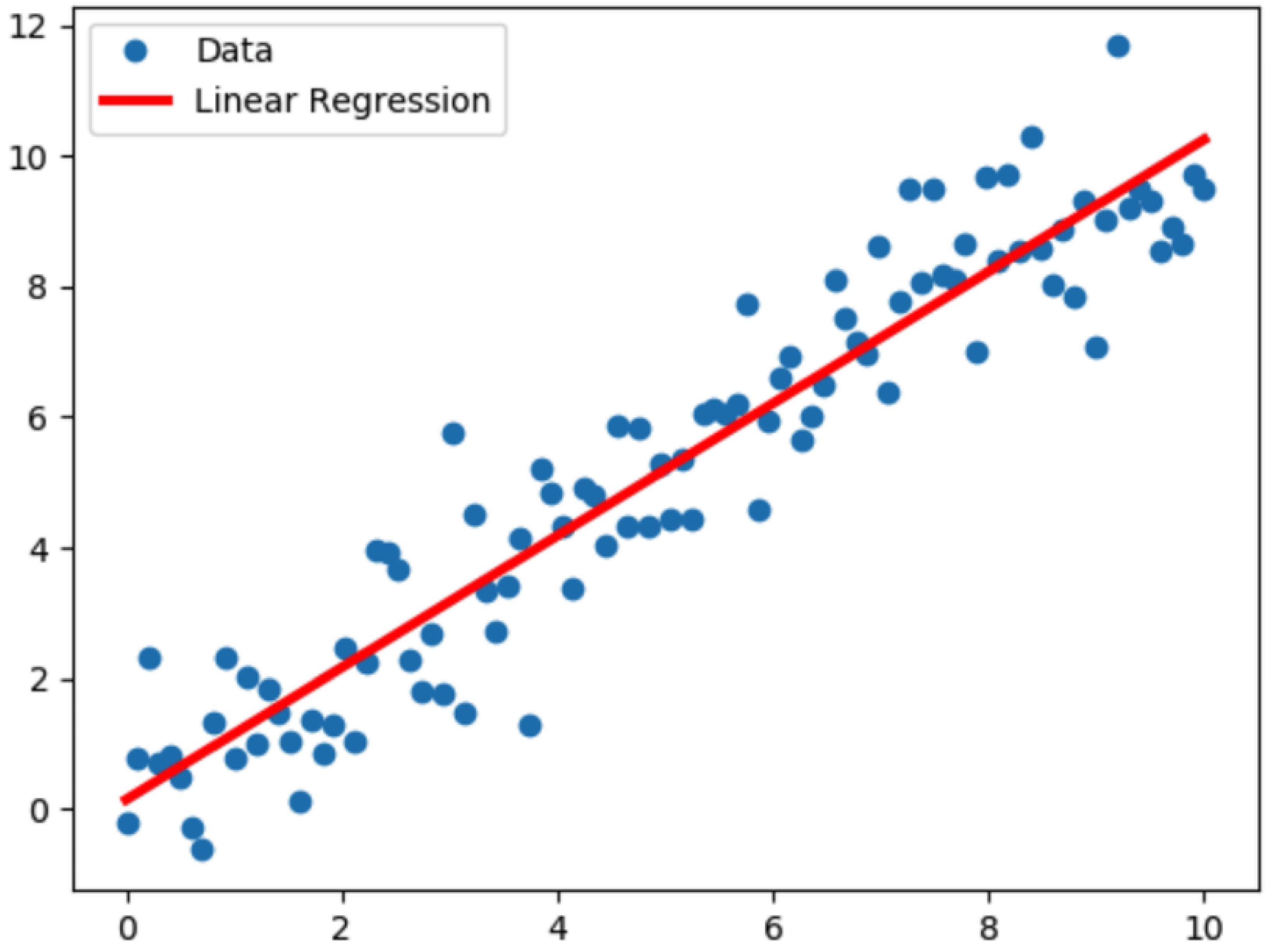


REGRESSION ALGORITHMS

- **Linear Regression**
- **Ridge Regression**
- **Lasso Regression**
- **KNN (Regression Friendly)**
- **Decision Trees (Regression Friendly)**

LINEAR REGRESSION

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_{p-1} X_{i,p-1} + \varepsilon_i$$



LINEAR REGRESSION

The Algorithm:

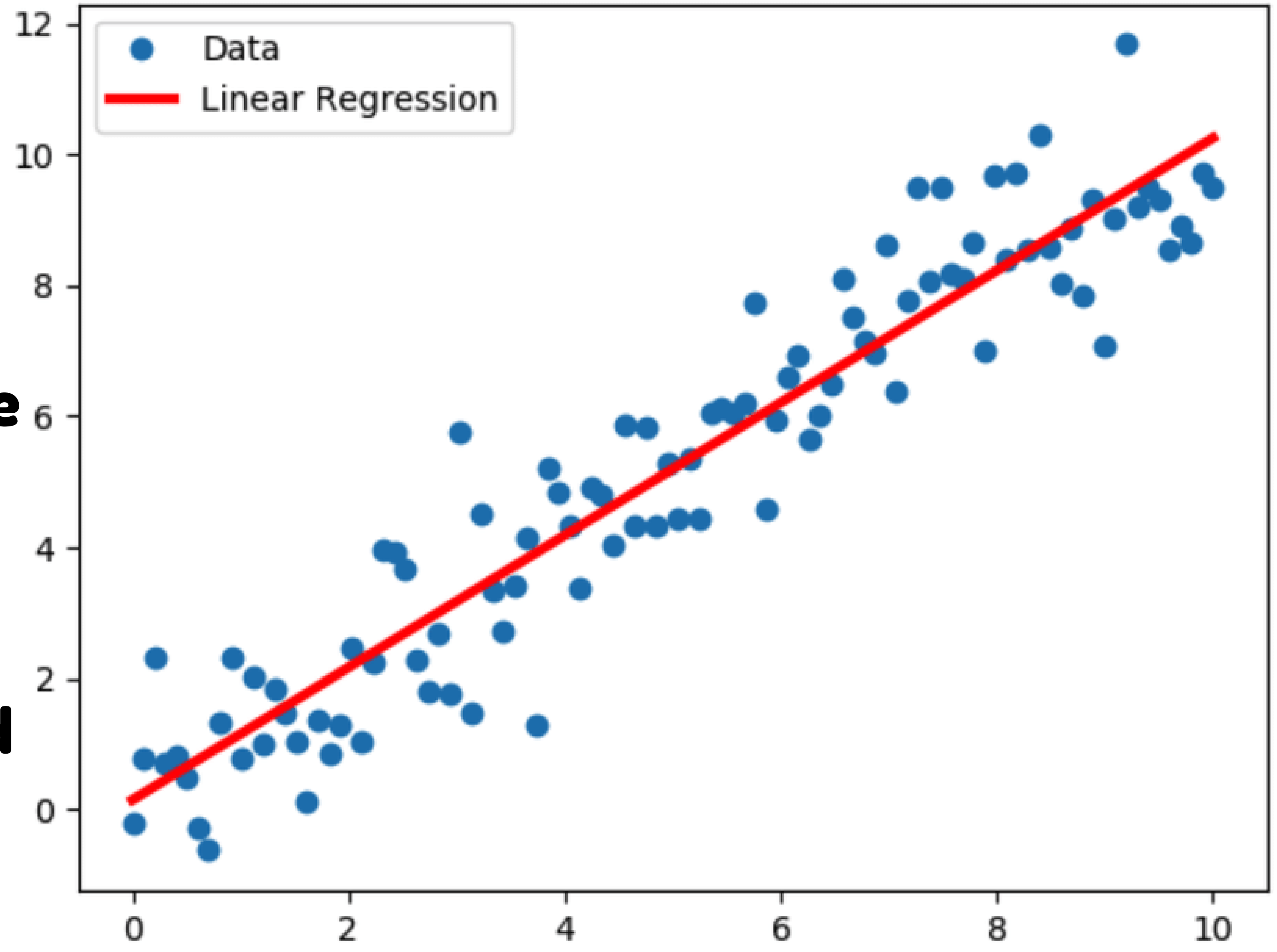
- **Identify the algorithm parameters: coefficients**

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_{p-1} X_{i,p-1} + \varepsilon_i$$

- **Optimize values for the coefficients that minimize the error**

$$\sum_{i=1}^{N} (y_i - \hat{y}_i)^2$$

- **Evaluate the error made between your model and predictions**



RIDGE REGRESSION

Exactly the same concept as linear regression (in fact this is a linear regression model), but with an additional constraint:

All Parameters will try to be optimized in order to be the smallest possible

$$Y_i = \beta_0 + \beta_1 X_{i1} + \beta_2 X_{i2} + \ldots + \beta_{p-1} X_{i,p-1} + \varepsilon_i$$

$$\sum_{i=1}^{N} (y_i - \hat{y}_i)^2 + \alpha \sum_{j=1}^{p} \beta_j^2$$

This means that the model will try to have each feature have as little impact as possible on the outcome, but still getting the best possible prediction

This is an example of a technique called Regularization which is built in order to explicitely avoid overfitting

RIDGE REGRESSION

When compared with the usual linear regression we should expect a lower score in the training dataset (less overfitting)

But then we should expect a higher test score since our model should not have so much overfitting

RIDGE REGRESSION

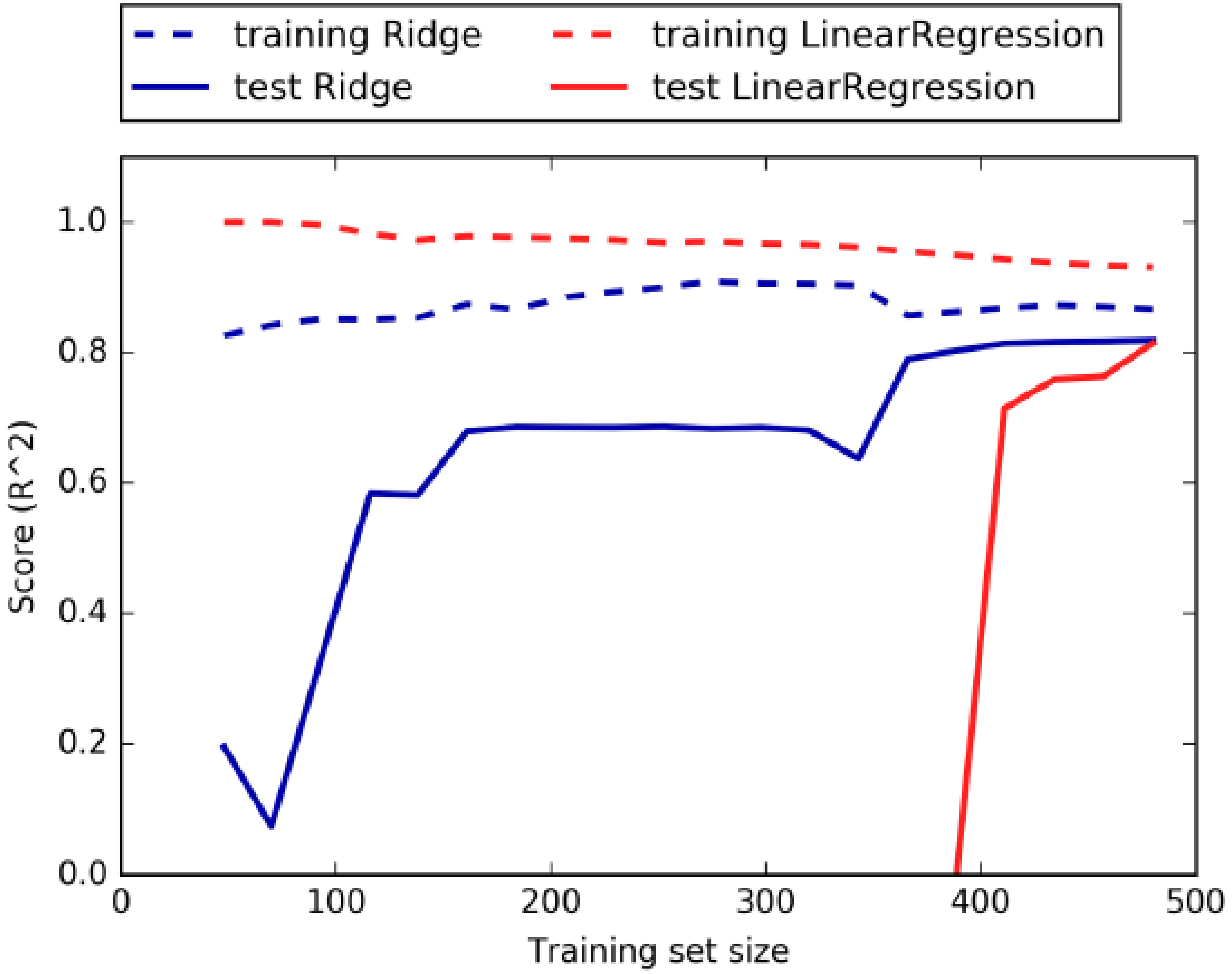
When compared with the usual linear regression we should expect a lower score in the training dataset (less overfitting)

But then we should expect a higher test score since our model should not have so much overfitting

Ridge therefore performs **a tardeoff between simplicity and the testing set performance**

Hyperparameter alpha controls how much this control on parameter magnitude is

RIDGE REGRESSION



LASSO REGRESSION

Whilst Ridge tries to minimize the magnitude of the parameters, it still keeps all the variables in its model

Lasso Regression on the other hand, will also perform regularization but allowing the coefficients to actually be zero thus "eliminating" features with negligible predictive power or consequence.

LASSO REGRESSION

This has incurs in the danger of making the model too simple and thus underfitting

However we can also tune the hyperparameter alpha to try to get the greatest amount of significant features

If we are able to obtain a similar or slightly higher test score using Lasso Regression , we should keep it as means we managed to achieve a model with similar performance using less parameters!

LINEAR REGRESSION - SUMMARY

+

Easily interpretable
Well known
Can model relatively complex phenomena with transformations

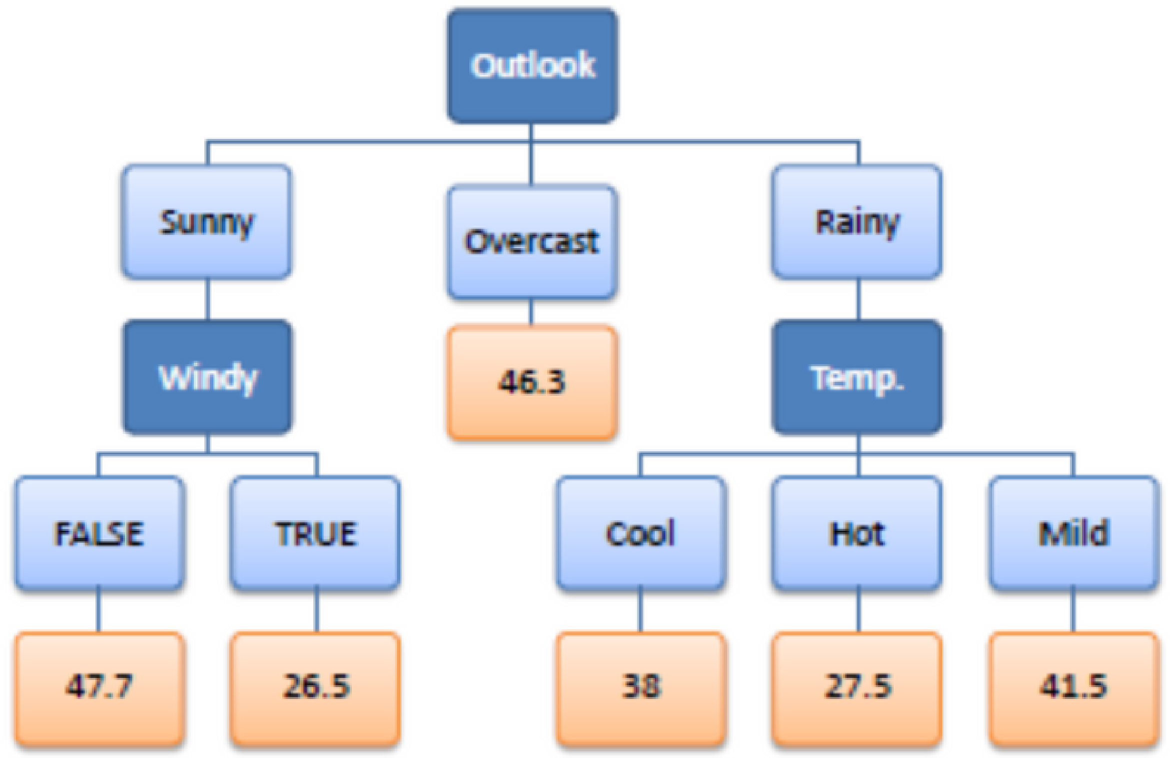
−

Must know about structure of relationships in data
Struggles with complex relationships

DECISION TREES

Takes a subset of training instances, and splits the data based on those features that give the most information
In multiple decision trees models, the average of each tree's result is returned
The 'deeper' the tree (more nodes) the more closely it follows the training data

Predictors				Target
Outlook	**Temp.**	**Humidity**	**Windy**	**Hours Played**
Rainy	Hot	High	False	26
Rainy	Hot	High	True	30
Overcast	Hot	High	False	48
Sunny	Mild	High	False	46
Sunny	Cool	Normal	False	62
Sunny	Cool	Normal	True	23
Overcast	Cool	Normal	True	43
Rainy	Mild	High	False	36
Rainy	Cool	Normal	False	38
Sunny	Mild	Normal	False	48
Rainy	Mild	Normal	True	48
Overcast	Mild	High	True	62
Overcast	Hot	Normal	False	44
Sunny	Mild	High	True	30



DECISION TREES - SUMMARY

+

Very robust to different types of features, including NaNs, categorical, numerical
Handles non-linearity
Robust to outliers
Can be stacked into arguably the most useful models in modern ML

−

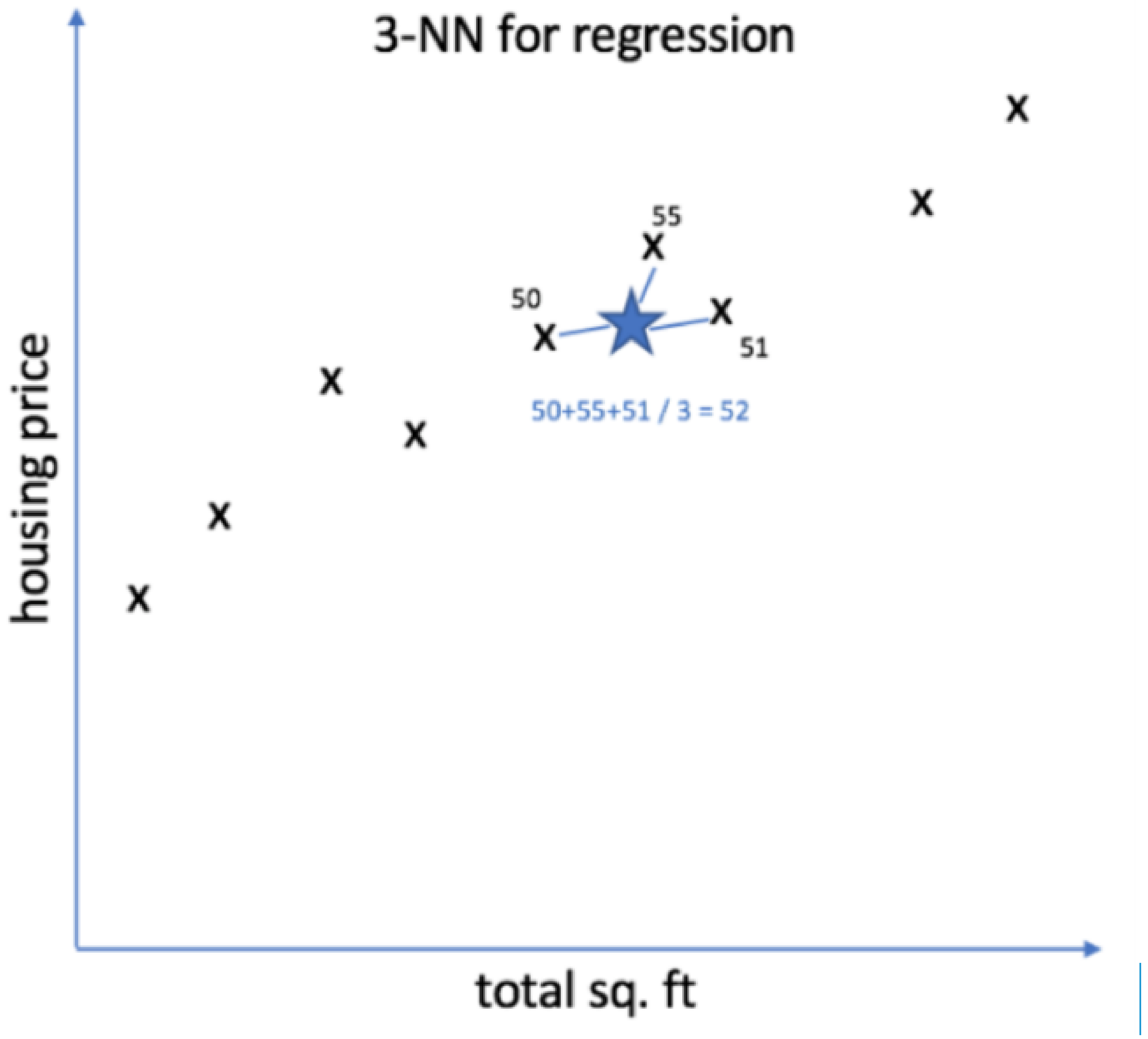
Very quick to overfit
Very "brittle" - few new observations can reconfigure the whole tree

KNN REGRESSION

Label for a new datapoint is assigned based on the mean of the outcome variable of K data points closest to the datapoint.

Average can also be weighted by distance.

Normalization of variables may be required



KNN REGRESSION - SUMMARY

+

No assumptions about data
Can be used to "seed" other methods, cheaply fill in missing values, etc

−

Requires all data to be stored in memory to compute
Can under perform with many variables
Very sensitive to scale

PERFORMANCE EVALUATION METRICS

The mean absolute error it is the average (mean) of the absolute value of the distance between predicted and actual values. It is a measure of absolute error: because of this, we do not know if the algorithm is overestimating or underestimating when it is incorrect.

Divide by the total number of data points

Predicted output value

Actual output value

$$MAE = \boxed{\frac{1}{n}} \sum \underbrace{\left| \boxed{y} - \boxed{\hat{y}} \right|}_{\text{The absolute value of the residual}}$$

Sum of

PERFORMANCE EVALUATION METRICS

Mean Squared Error is a more common metric. The squaring of the value penalizes more larger errors. However, it can also mean that a relatively small number of incorrect outlier predictions can have a disproportionately large negative effect on score. Sometimes it can make sense to take the square room of MSE to make the result "comparable" with the scale of individual observations, just like when we were comparing variance and std.

$$MSE = \frac{1}{n}\Sigma\underbrace{\left(y-\hat{y}\right)^2}_{\text{The square of the difference between actual and predicted}}$$

PERFORMANCE EVALUATION METRICS

R^2 is the proportion of the total variance seen in data (denominator) that is explained by our predictions. If our predictions are dead on, the numerator is 0 and our R^2 is 1. If our predictions are so bad that they are functionally equivalent to answering "the mean", our numerator and denominator are equal and R^2 equals 0.

$$R^2 = 1 - \frac{\sum_{i=0}^{samples-1} (y_i - \hat{y}_i)^2}{\sum_{i=0}^{samples-1} (y_i - \bar{y})^2}$$

ANY QUESTIONS ?

PERFORMANCE EVALUATION METRICS - ROC CURVE

It is a plot of the false positive rate (x-axis) versus the true positive rate (y-axis) for a number of different candidate threshold values between 0.0 and 1.0. Put another way, it plots the false alarm rate versus the hit rate.