Classification

Classification

- Binary
- Multiclass

Classification problem

Outlook	Temperature	Humidity	Windy	Play golf
Rainy	Hot	High	False	No
Rainy	Hot	High	True	No
Overcast	Hot	High	False	Yes
Sunny	Mil	High	False	Yes
Sunny	Cool	Normal	False	Yes
Sunny	Cool	Normal	True	No
Overcast	Cool	Normal	True	Yes
Rainy	Mild	High	False	No
Rainy	Cool	Normal	False	Yes

Classification predictions

Play golf	Model
No	No
No	Yes
Yes	Yes
Yes	Yes
Yes	No
No	No
Yes	Yes
No	Yes
Yes	Yes

Classification - Confusion matrix

Play golf	Model
No	No
No	Yes
Yes	Yes
Yes	Yes
Yes	No
No	No
Yes	Yes
No	Yes
Yes	Yes

		Predicted class	
		Positive	Negative
True class	Positive		
	Negative		

Classification - Confusion matrix

Play golf	Model
No	No
No	Yes
Yes	Yes
Yes	Yes
Yes	No
No	No
Yes	Yes
No	Yes
Yes	Yes

		Predicted class	
		Positive	Negative
True class	Positive	4	
	Negative		

Classification - Confusion matrix

Play golf	Model
No	No
No	Yes
Yes	Yes
Yes	Yes
Yes	No
No	No
Yes	Yes
No	Yes
Yes	Yes

		Predicted class	
		Positive	Negative
True class	Positive	4	1
	Negative		

Classification - Confusion matrix

Play golf	Model
No	No
No	Yes
Yes	Yes
Yes	Yes
Yes	No
No	No
Yes	Yes
No	Yes
Yes	Yes

		Predicted class	
		Positive	**Negative**
True class	**Positive**	4	1
	Negative	2	

Classification - Confusion matrix

Play golf	Model
No	No
No	Yes
Yes	Yes
Yes	Yes
Yes	No
No	No
Yes	Yes
No	Yes
Yes	Yes

		Predicted class	
		Positive	**Negative**
True class	**Positive**	4	1
	Negative	2	2

Classification - Confusion matrix

Play golf	Model
No	No
No	Yes
Yes	Yes
Yes	Yes
Yes	No
No	No
Yes	Yes
No	Yes
Yes	Yes

		Predicted class	
		Positive	**Negative**
True class	**Positive**	TP = 4	FN = 1
	Negative	FP = 2	TN = 2

Classification - Metrics

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}$$

		Predicted class	
		Positive	**Negative**
True class	**Positive**	TP = 4	FN = 1
	Negative	FP = 2	TN = 2

Classification - Metrics

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} = \frac{4 + 2}{4 + 2 + 2 + 1} = \frac{6}{9} = 66.7\%$$

		Predicted class	
		Positive	**Negative**
True class	**Positive**	TP = 4	FN = 1
	Negative	FP = 2	TN = 2

Classification - Metrics

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} = \frac{4 + 2}{4 + 2 + 2 + 1} = \frac{6}{9} = 66.7\%$$

Not always works well!

		Predicted class	
		Positive	**Negative**
True class	**Positive**	TP = 4	FN = 1
	Negative	FP = 2	TN = 2

Classification - Metrics

100 patients:

  99 without cancer

  1 with cancer

Great model idea:

Always predict no cancer

Classification - Metrics

100 patients:
    99 without cancer
    1 with cancer

Great model idea:
    Always predict no cancer

		Predicted class	
		Positive	**Negative**
True class	**Positive**	TP = 0	FN = 1
	Negative	FP = 0	TN = 99

Classification - Metrics

100 patients:
- 99 without cancer
- 1 with cancer

Great model idea:
Always predict no cancer

		Predicted class	
		Positive	**Negative**
True class	**Positive**	TP = 0	FN = 1
	Negative	FP = 0	TN = 99

$$\text{Accuracy} = \frac{99}{100} = 99\%$$

Classification - Metrics

$$\text{Recall} = \text{TPR} = \frac{TP}{TP + FN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

Classification - Metrics: Play Golf

$$\text{Recall} = \text{TPR} = \frac{TP}{TP + FN}$$

		Predicted class	
		Positive	**Negative**
True class	**Positive**	TP = 4	FN = 1
	Negative	FP = 2	TN = 2

Classification - Metrics: Play Golf

$$\text{Recall} = \text{TPR} = \frac{TP}{TP + FN} = \frac{4}{4+1} = \frac{4}{5} = 80\%$$

		Predicted class	
		Positive	**Negative**
True class	**Positive**	TP = 4	FN = 1
	Negative	FP = 2	TN = 2

Classification - Metrics: Play Golf

$$\text{Precision} = \frac{TP}{TP + FP}$$

		Predicted class	
		Positive	**Negative**
True class	**Positive**	TP = 4	FN = 1
	Negative	FP = 2	TN = 2

Classification - Metrics: Play Golf

$$\text{Precision} = \frac{TP}{TP + FP} = \frac{4}{4+2} = \frac{4}{6} = 66.7\%$$

		Predicted class	
		Positive	**Negative**
True class	**Positive**	TP = 4	FN = 1
	Negative	FP = 2	TN = 2

Classification - Metrics: Play Golf

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} = \frac{4 + 2}{4 + 2 + 2 + 1} = \frac{6}{9} = 66.7\%$$

$$\text{Recall} = \text{TPR} = \frac{TP}{TP + FN} = \frac{4}{4 + 1} = \frac{4}{5} = 80\%$$

$$\text{Precision} = \frac{TP}{TP + FP} = \frac{4}{4 + 2} = \frac{4}{6} = 66.7\%$$

Classification - Metrics: Cancer

100 patients:

 99 without cancer

 1 with cancer

Great model idea:

 Always predict no cancer

		Predicted class	
		Positive	**Negative**
True class	**Positive**	TP = 0	FN = 1
	Negative	FP = 0	TN = 99

Accuracy = 99%

Classification - Metrics: Cancer

100 patients:
- 99 without cancer
- 1 with cancer

Great model idea:
Always predict no cancer

		Predicted class	
		Positive	**Negative**
True class	**Positive**	TP = 0	FN = 1
	Negative	FP = 0	TN = 99

Accuracy = 99%

Recall = 0%

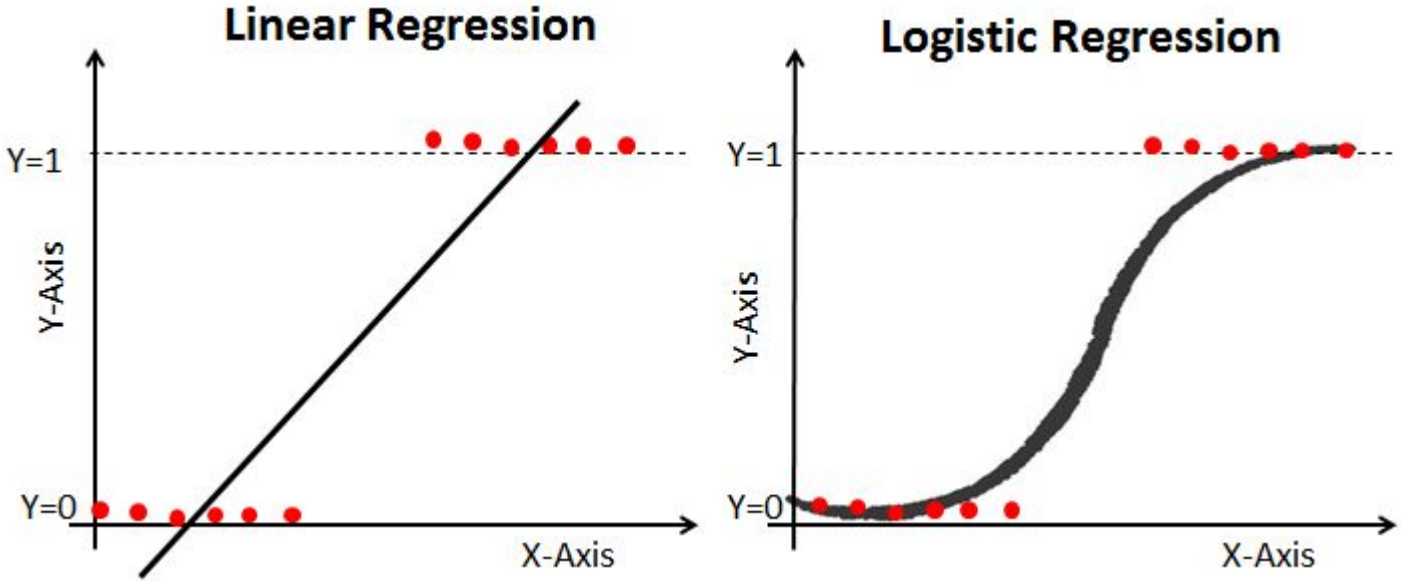
Precision = NaN

Classification - More Metrics

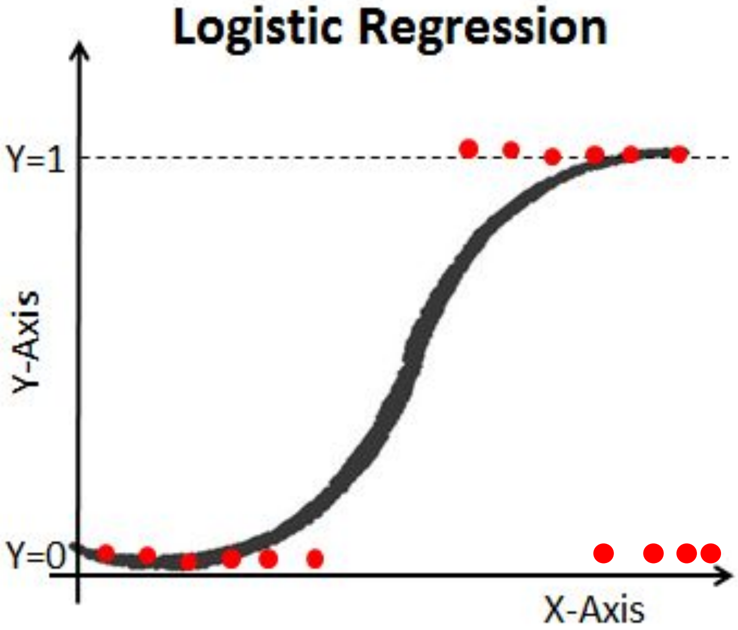
https://en.wikipedia.org/wiki/Precision_and_recall

Logistic Regression

Logistic Regression



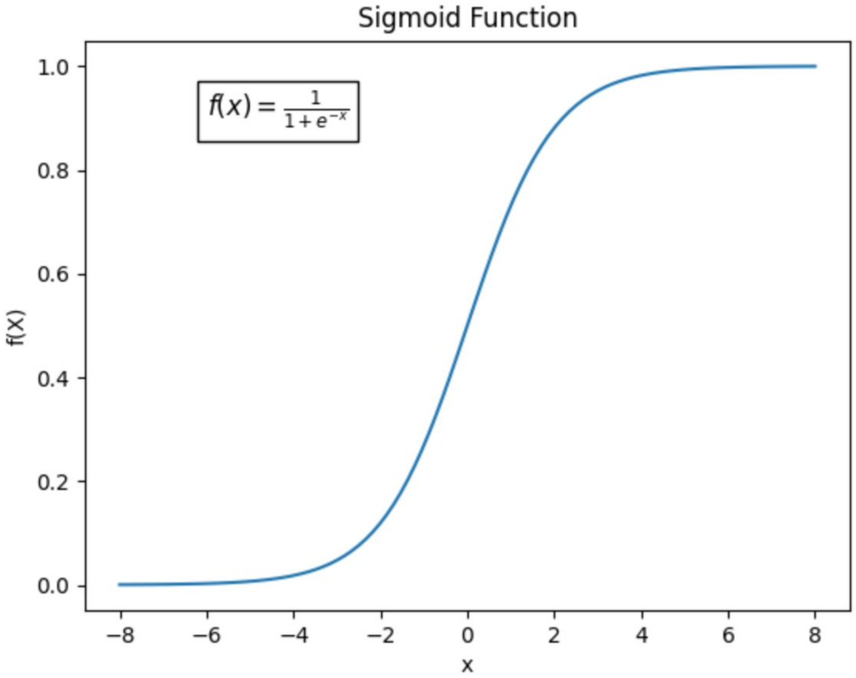
Logistic Regression – Non-linear



Log-Loss

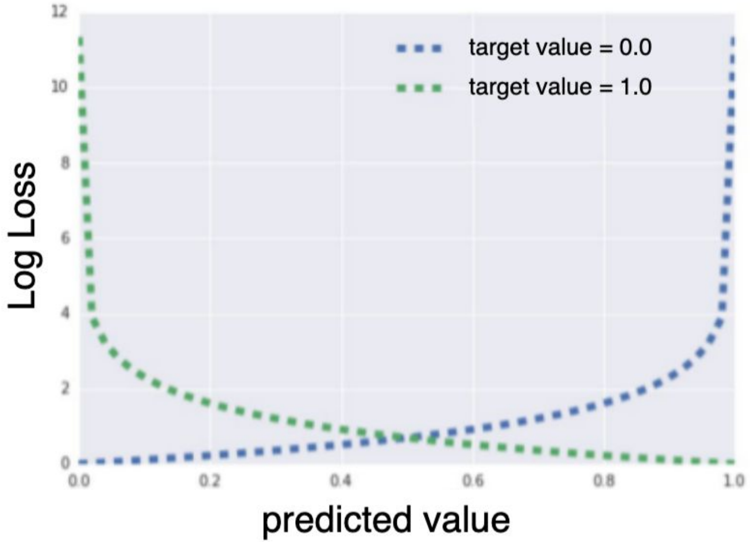
$$P(y = 1|X) = \text{sigmoid}(z) = \frac{1}{1 + e^{-z}}$$

$$z = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \ldots + \hat{\beta}_k x_k$$



Minimize Log-Loss

$$LogLoss = \sum_{(x,y)\in D} -y\,log(y') - (1-y)\,log(1-y')$$



Logistic Regression: Important Hyperparams.

- penalty: None, l1, l2, elasticnet
- C:
  - Regularization parameter.
  - Only used when penalty is not None.
  - Large C can lead to overfitting.
  - Small C can lead to underfitting.
- solver:
  - Algorithm to use for optimization of the model.
  - Different solvers can handle different types of data.
  - Different solvers have different performance characteristics.
  - Check the documentation for specific information.

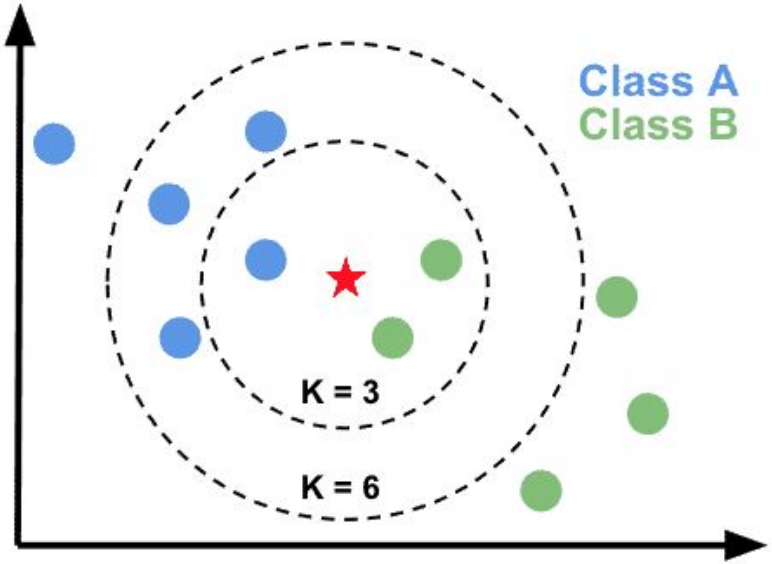
$$J(w) = C \cdot \text{LogLoss}(w) + \text{EN(w)}$$

Logistic Regression: Penalty & C

- L1: $J(w) = C \cdot \text{LogLoss}(w) + \sum_{j=1}^{n} |w_j|$
- L2: $J(w) = C \cdot \text{LogLoss}(w) + \frac{1}{2} \sum_{j=1}^{n} w_j^2$
- EN: $J(w) = C \cdot \text{LogLoss}(w) + \text{EN(w)}$

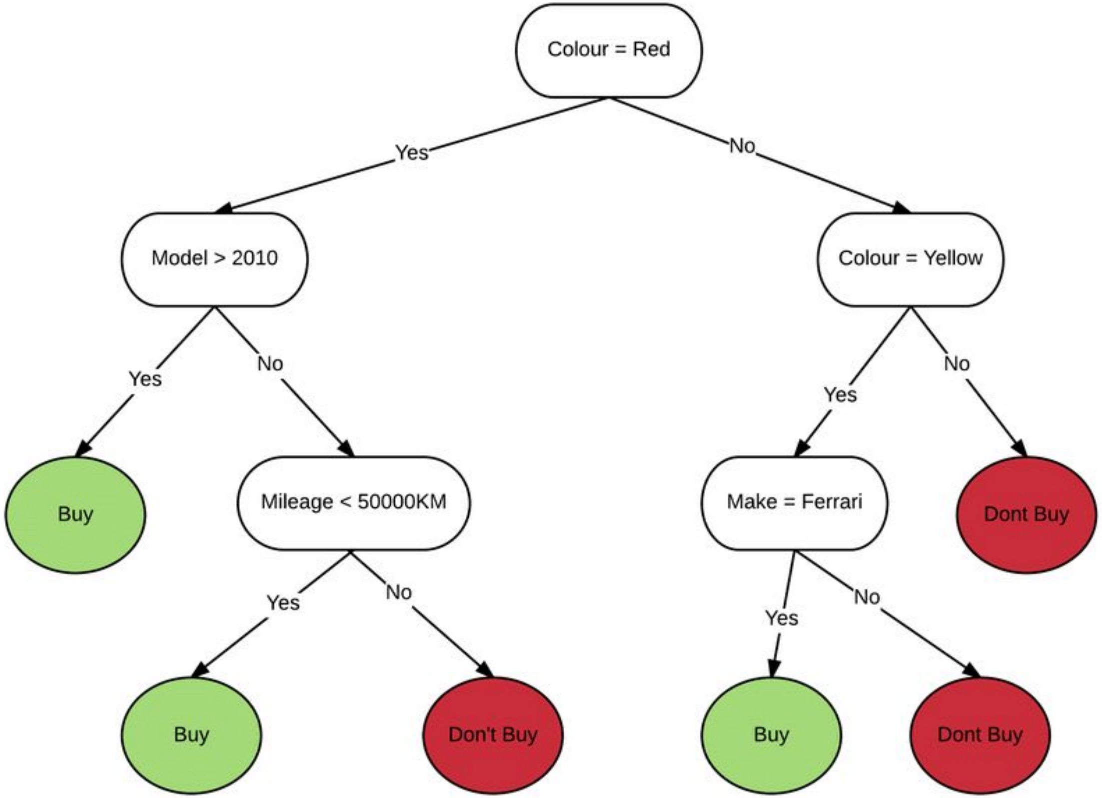
K-Nearest Neighbors

KNN



Decision Tree

DT


Colour = Red
Yes
No
Model > 2010
Colour = Yellow
Yes
No
Yes
No
Buy
Mileage < 50000KM
Make = Ferrari
Dont Buy
Yes
No
Yes
No
Buy
Don't Buy
Buy
Dont Buy

Impurity Criterion

Gini Index

$$I_G = 1 - \sum_{j=1}^{c} p_j^2$$

p_j: proportion of the samples that belongs to class c for a particular node

Entropy

$$I_H = - \sum_{j=1}^{c} p_j log_2(p_j)$$

p_j: proportion of the samples that belongs to class c for a particular node.

*This is the the definition of entropy for all non-empty classes ($p \neq 0$). The entropy is 0 if all samples at a node belong to the same class.

Gini Gain

$$Gini_{gain} = Gini_{parent} - \left(\frac{N_{left}}{N_{total}} Gini_{left} + \frac{N_{right}}{N_{total}} Gini_{right} \right)$$