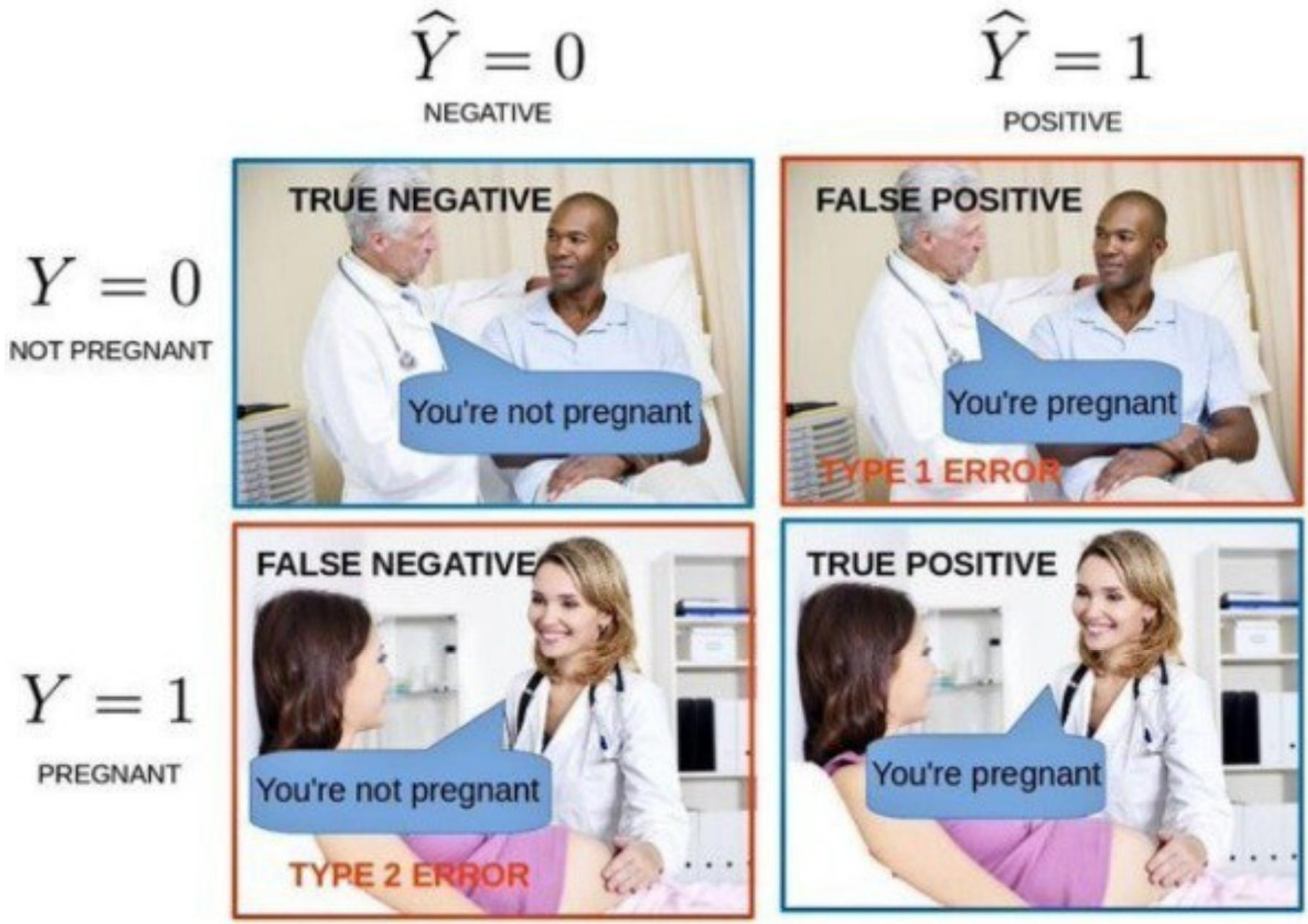
Unione europea
Fondo sociale europeo

Regione Emilia-Romagna

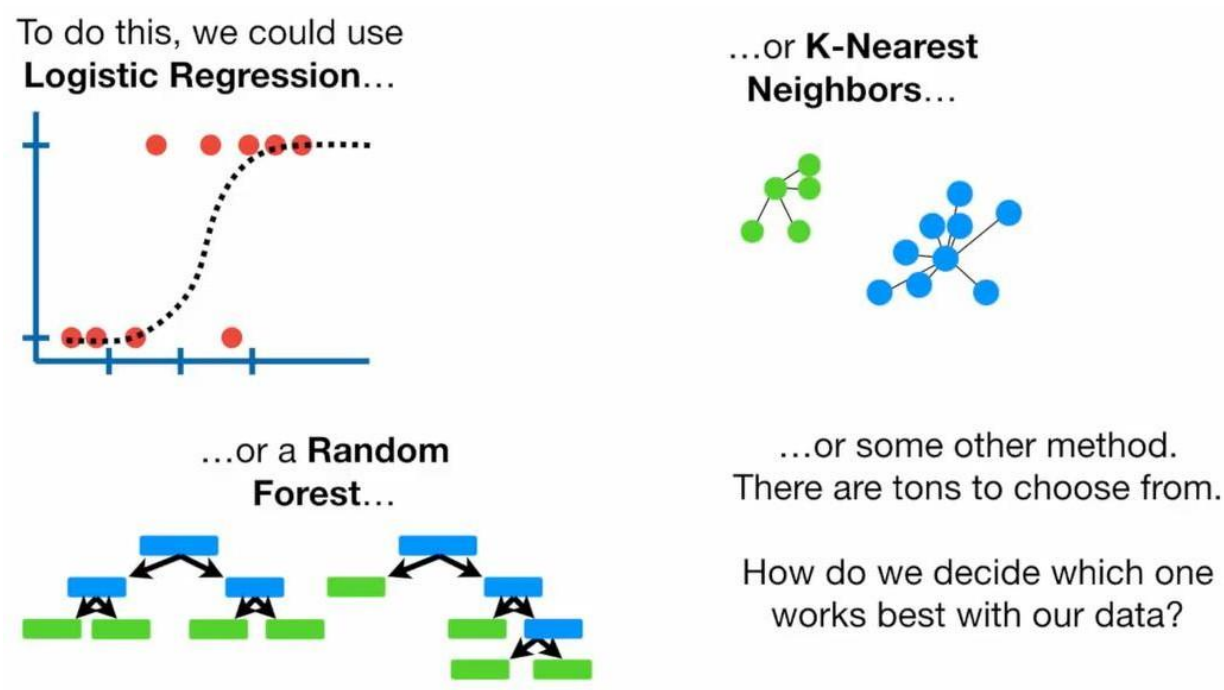
RETE POLITECNICA
La tecnica per crescere

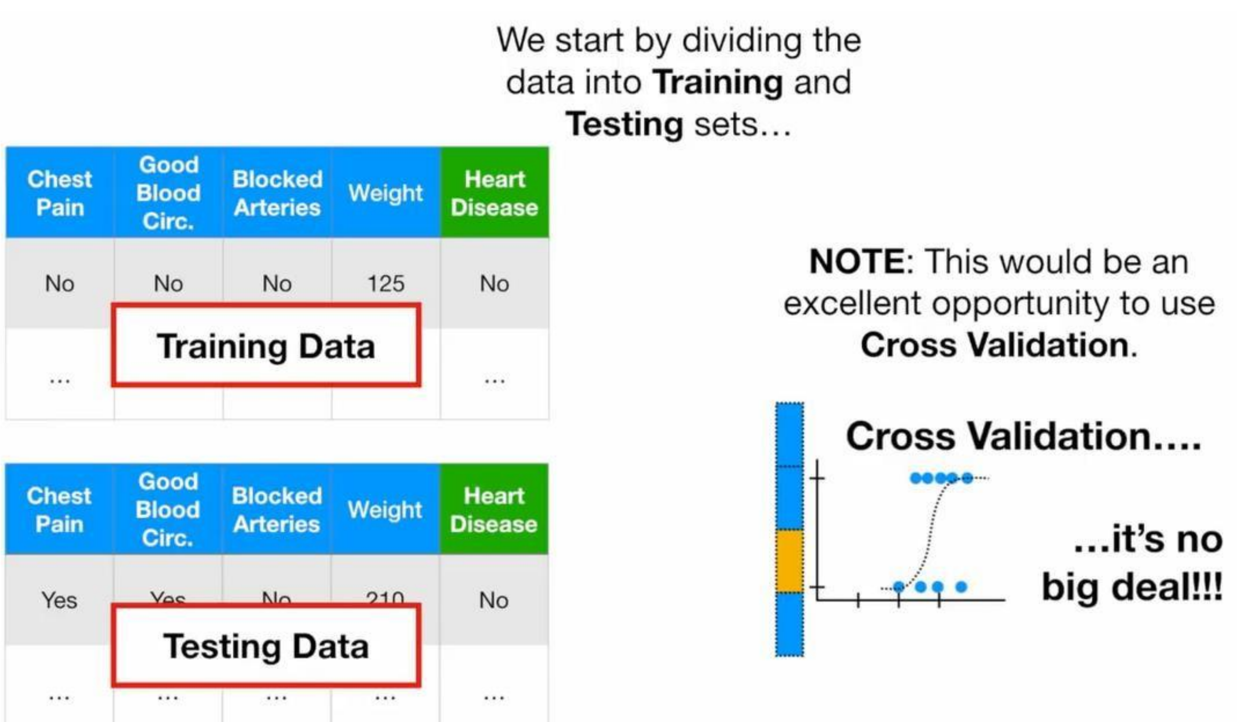
Materiale didattico per partecipante al corso “**TECNICO ESPERTO NELL’ANALISI E NELLA VISUALIZZAZIONE DEI DATI**” – Rif.P.A. 2021-15998/RER – approvata con DGR n. 1263 del 02/08/2021 di IFOA – Istituto Formazione Operatori Aziendali

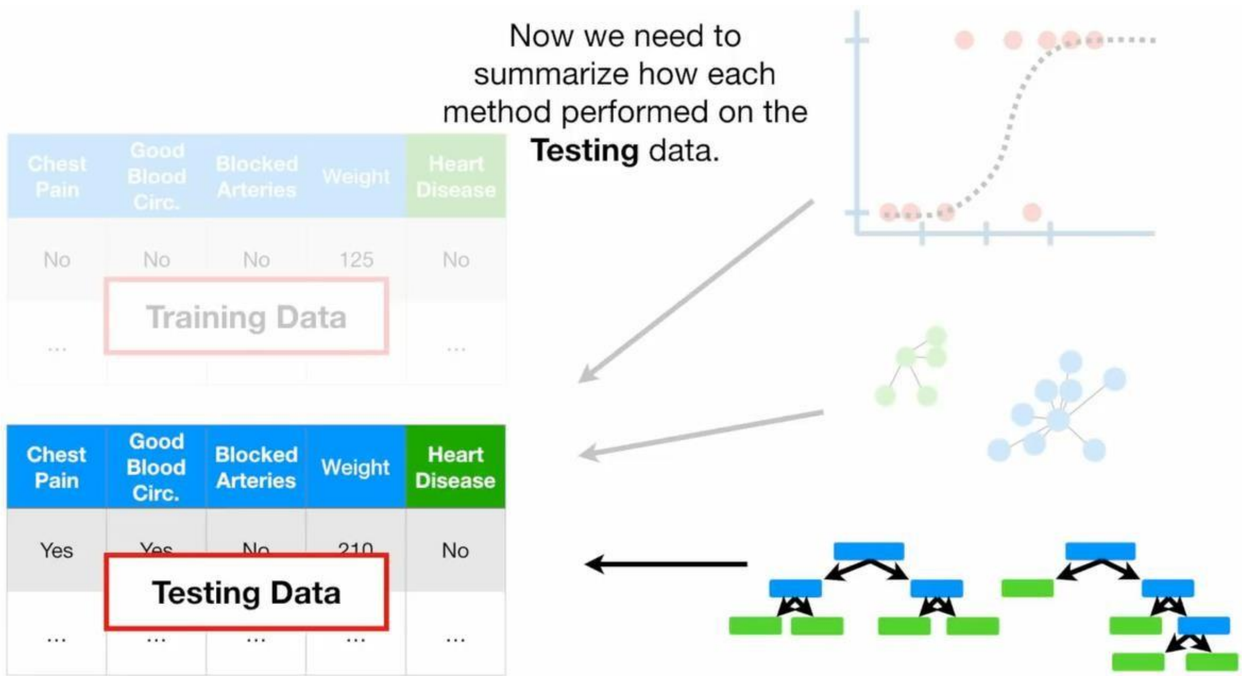


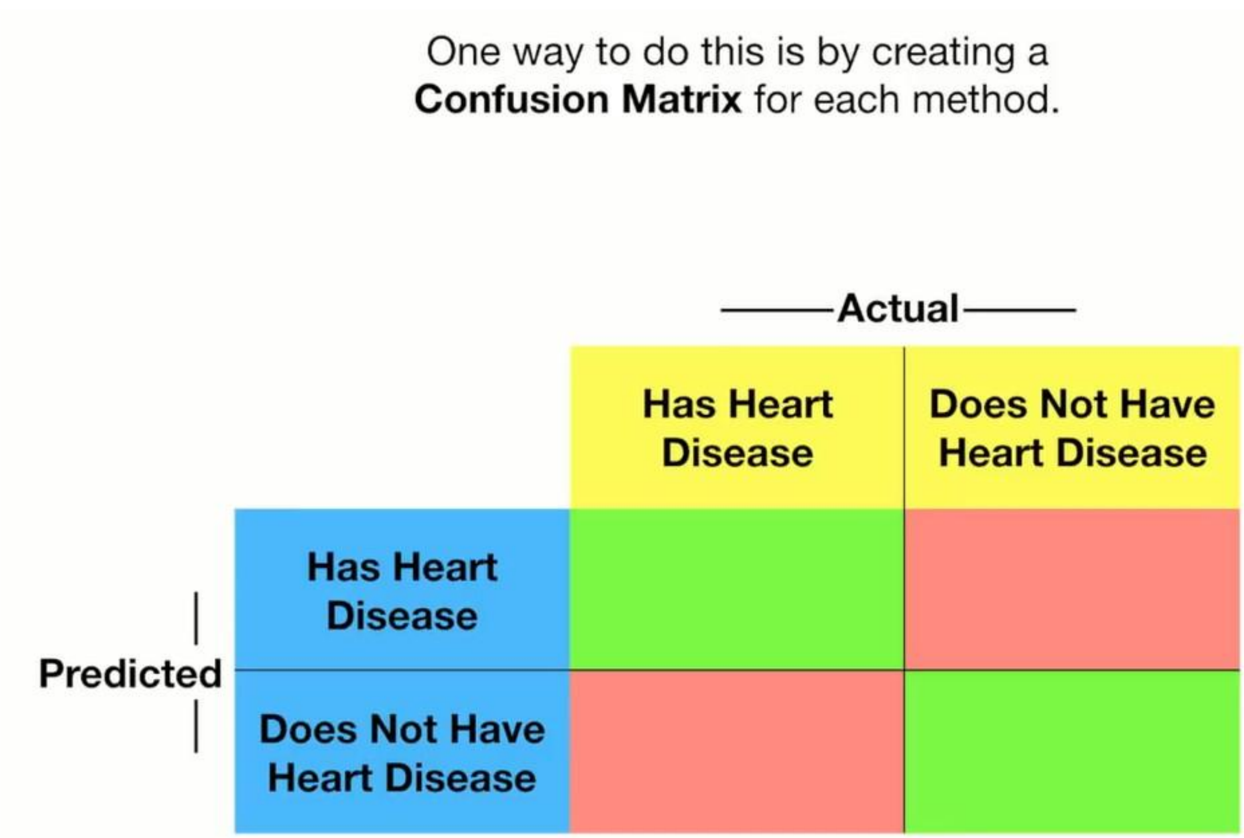
$\widehat{Y} = 0$
NEGATIVE
$\widehat{Y} = 1$
POSITIVE
$Y = 0$
NOT PREGNANT
TRUE NEGATIVE
You're not pregnant
FALSE POSITIVE
You're pregnant
TYPE 1 ERROR
$Y = 1$
PREGNANT
FALSE NEGATIVE
You're not pregnant
TYPE 2 ERROR
TRUE POSITIVE
You're pregnant

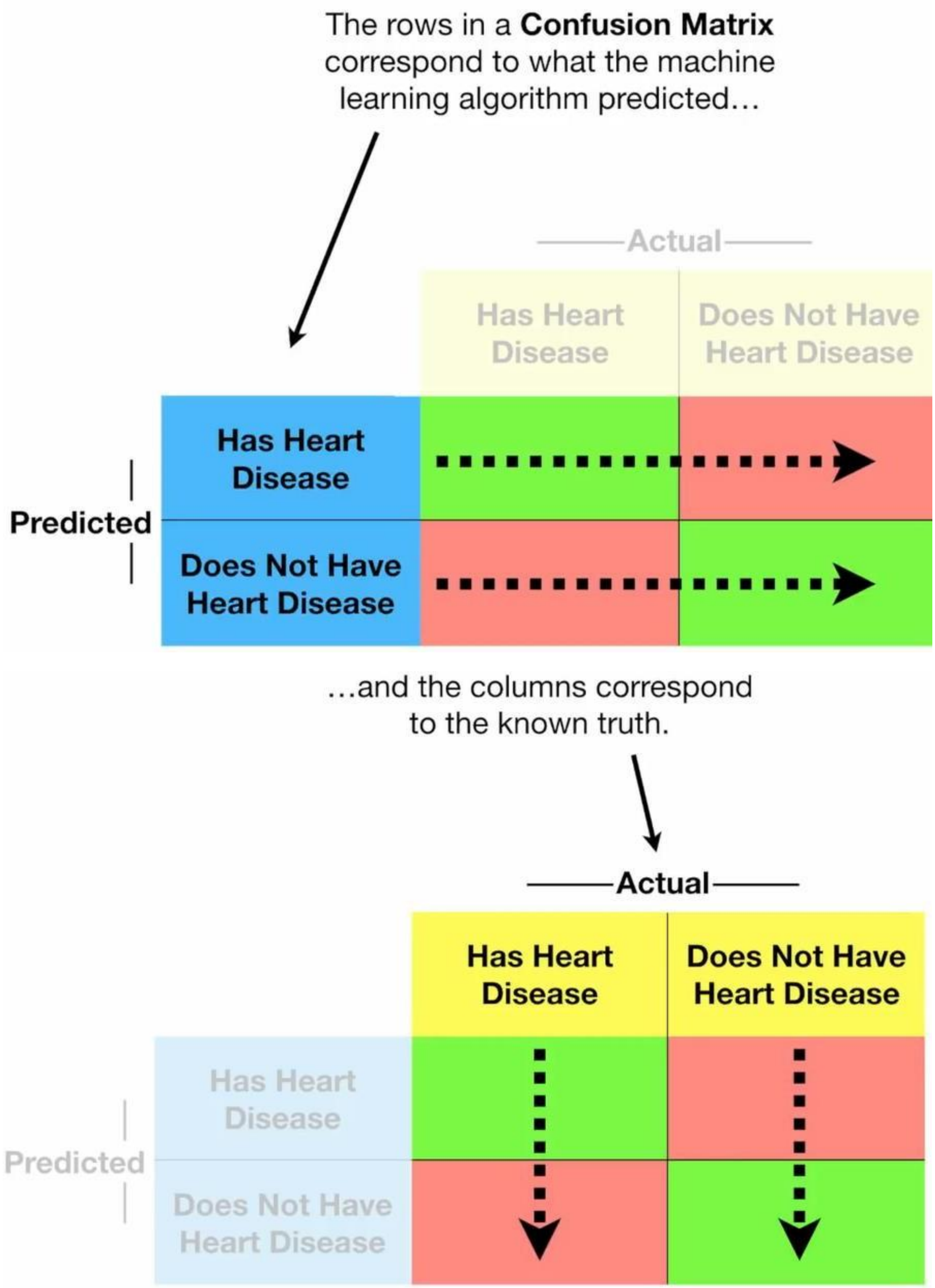
CONFUSION MATRIX



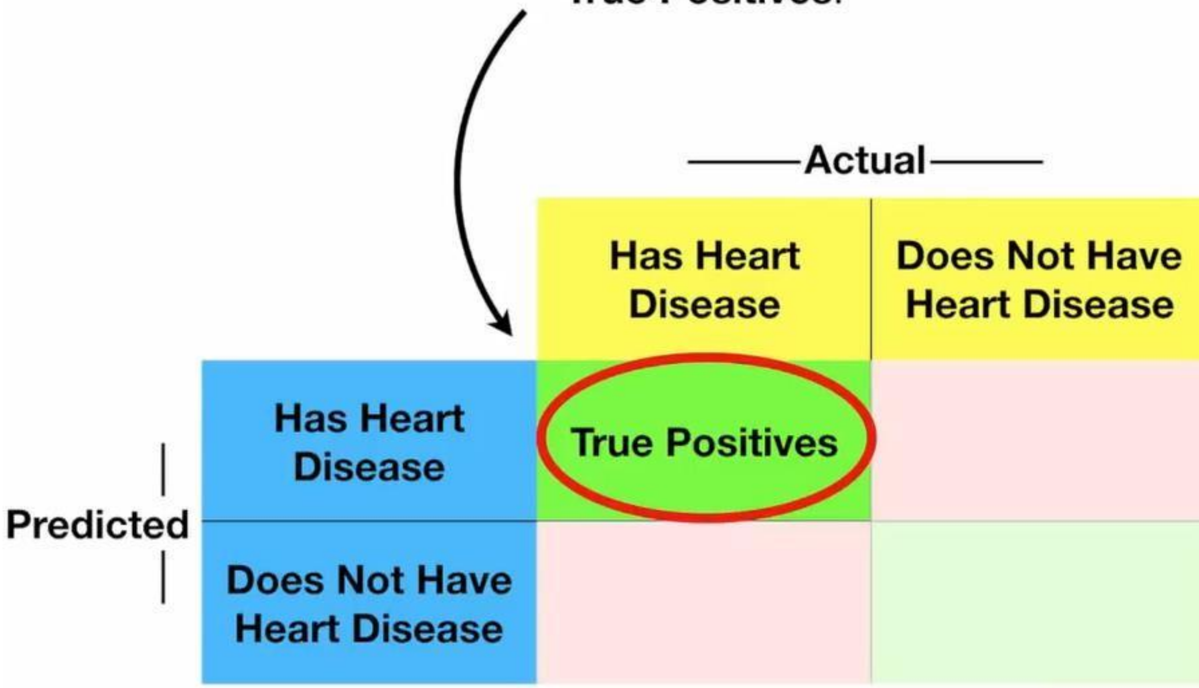



Now we need to summarize how each method performed on the **Testing** data.
Chest Pain
Good Blood Circ.
Blocked Arteries
Weight
Heart Disease
No
No
No
125
No
Training Data
Yes
Yes
No
210
No
Testing Data

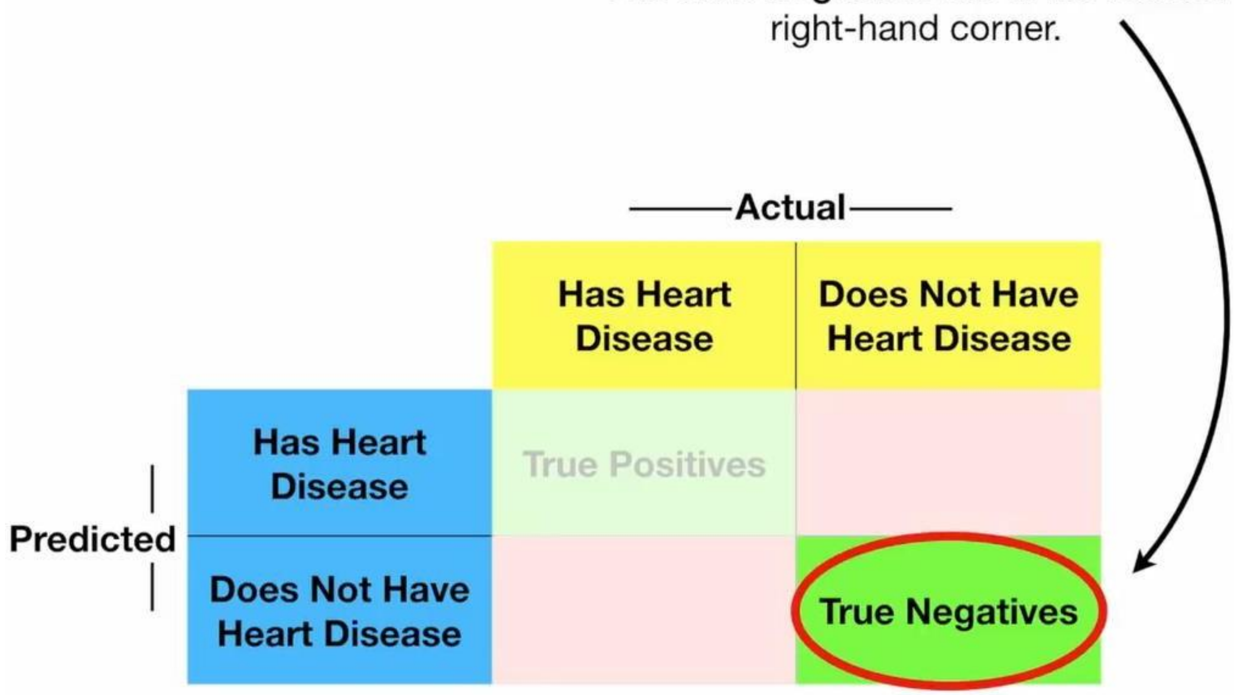

One way to do this is by creating a **Confusion Matrix** for each method.
Actual
Has Heart Disease
Does Not Have Heart Disease
Predicted
Has Heart Disease
Does Not Have Heart Disease


The rows in a **Confusion Matrix** correspond to what the machine learning algorithm predicted...
Actual
Has Heart Disease
Does Not Have Heart Disease
Predicted
Has Heart Disease
Does Not Have Heart Disease
...and the columns correspond to the known truth.
Actual
Has Heart Disease
Does Not Have Heart Disease
Predicted
Has Heart Disease
Does Not Have Heart Disease

…then the top left corner contains
True Positives.

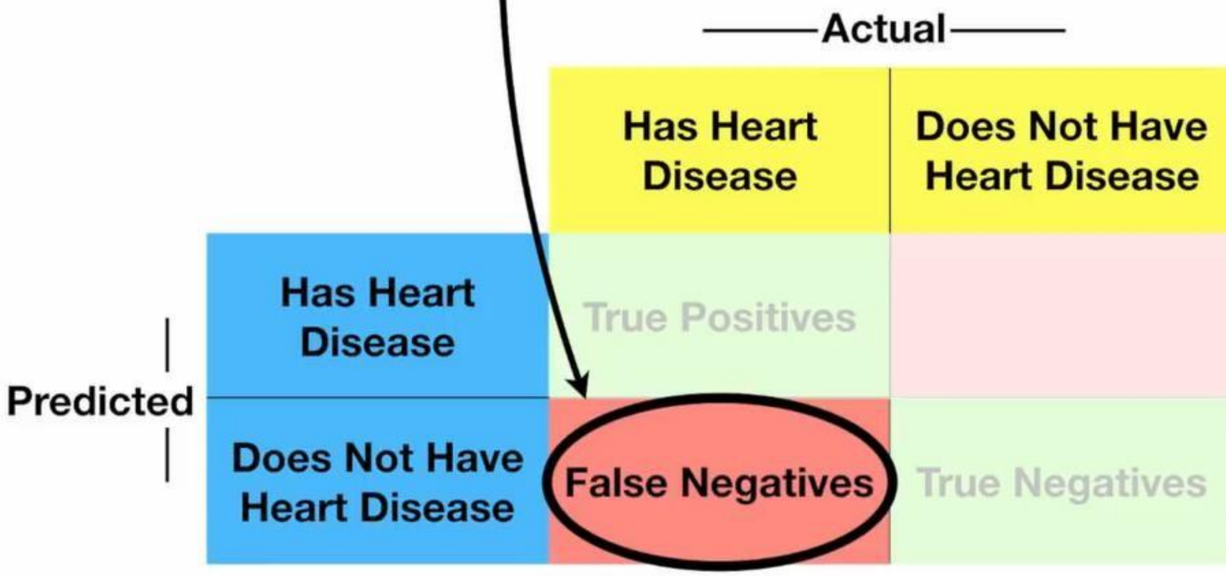


The **True Negatives** are in the bottom right-hand corner.

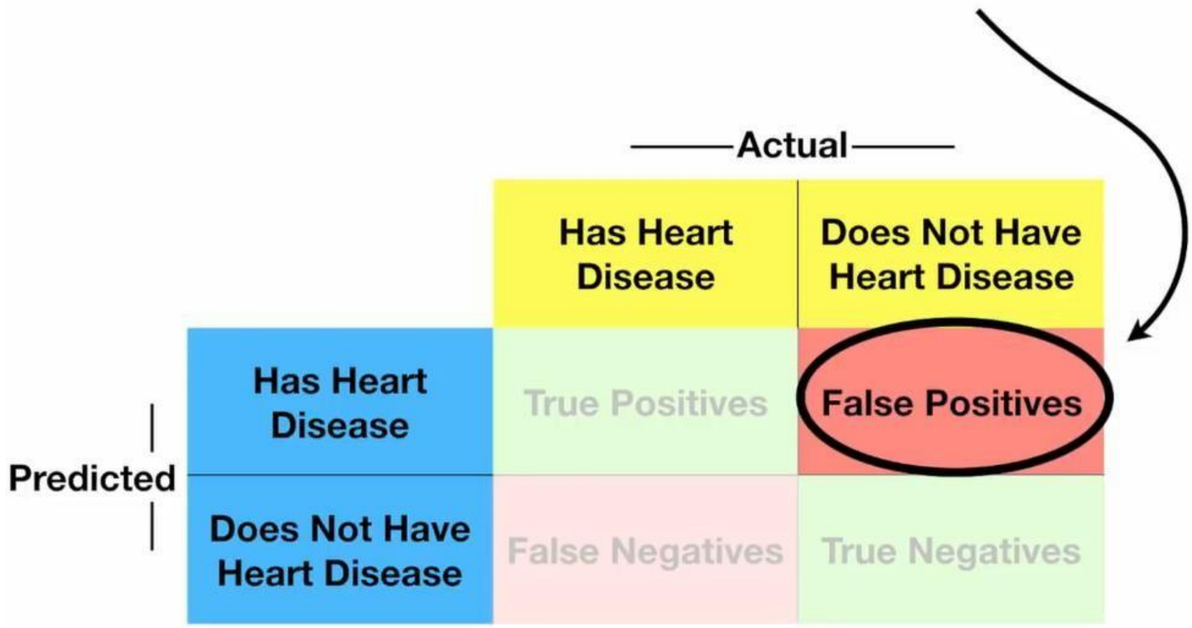


These are the patients that *did not have heart disease* that were correctly identified by the algorithm.

The bottom left-hand corner contains the **False Negatives**...



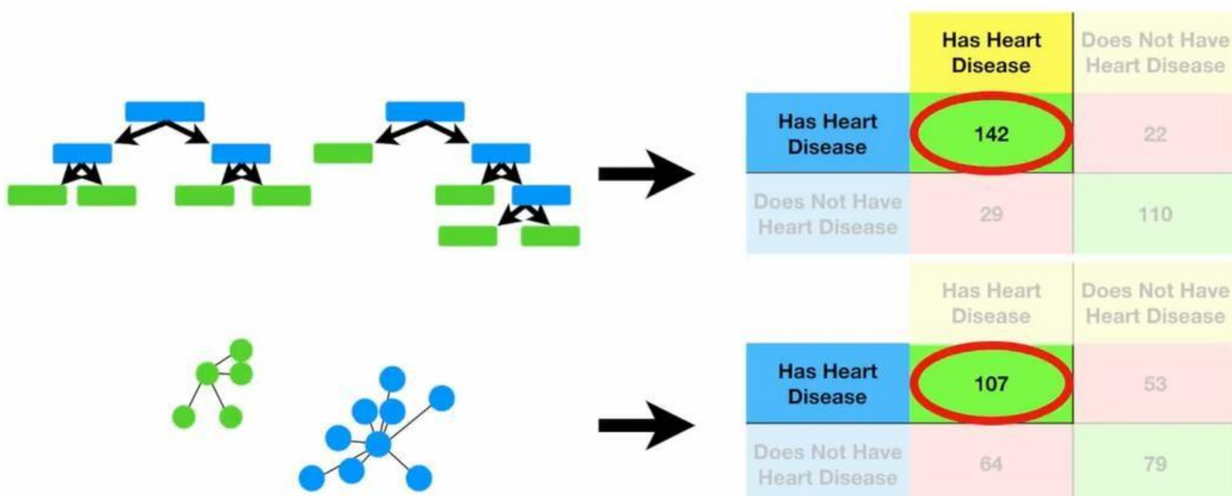
Lastly, the top right-hand corner contains the **False Positives**...



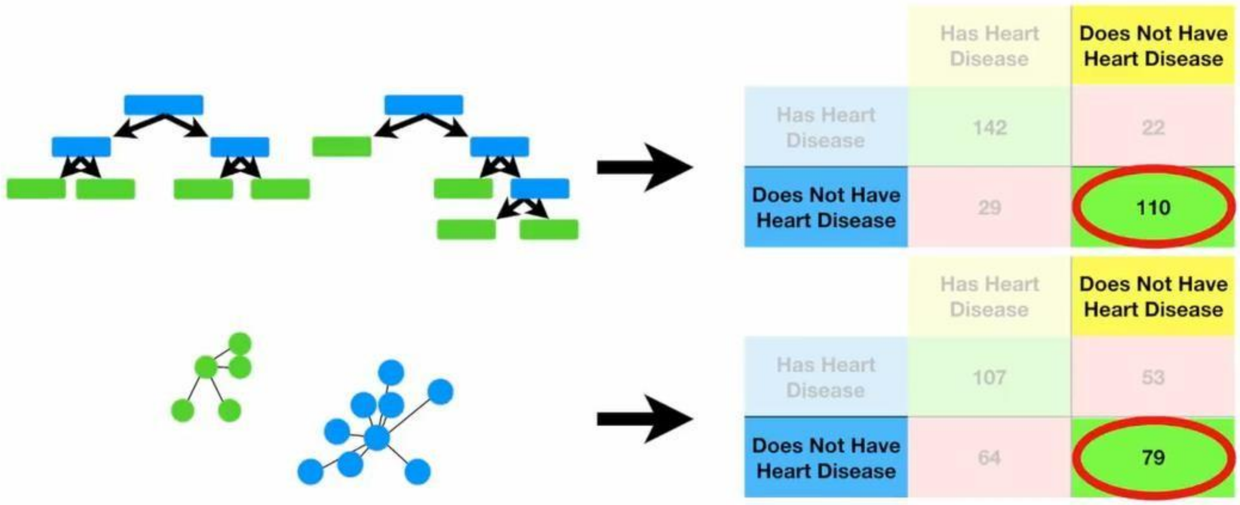
False Positives are patients that do not have heart disease, but the algorithm says they do.

…and the algorithm misclassified **22** patients that *did not* have heart disease by saying that they *did* (**False Positives**).

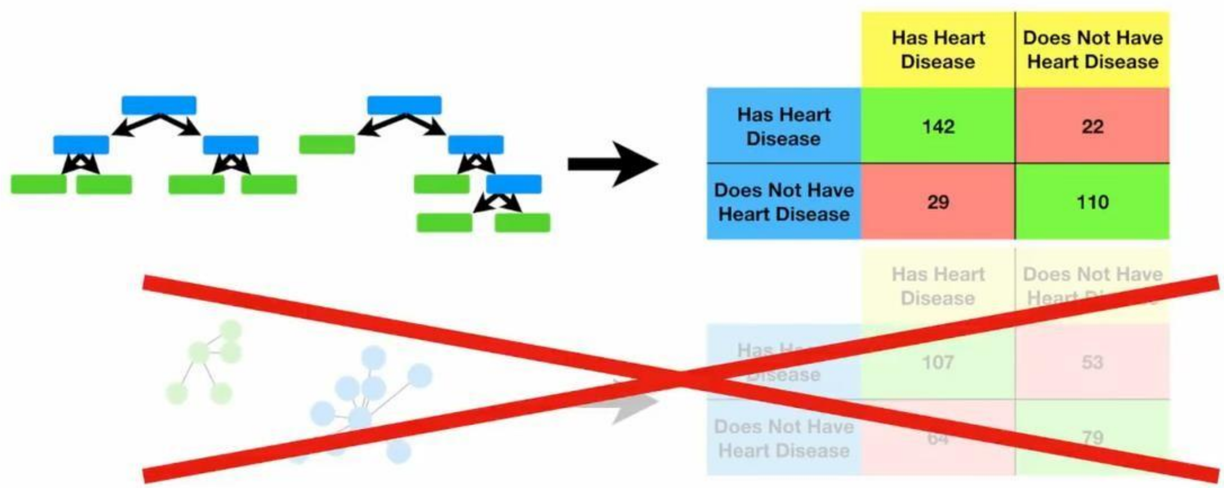
		Actual	
		Has Heart Disease	**Does Not Have Heart Disease**
Predicted	**Has Heart Disease**	142	**22**
	Does Not Have Heart Disease	**29**	110



K-Nearest Neighbors was worse than the **Random Forest** at predicting patients *with* Heart Disease (**107** vs **142**)…



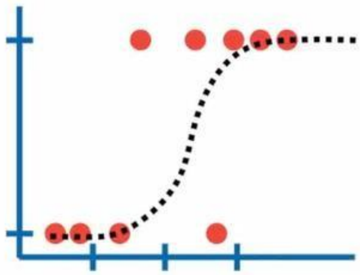
...and worse at predicting patients *without* Heart Disease (**79** vs **110**)...



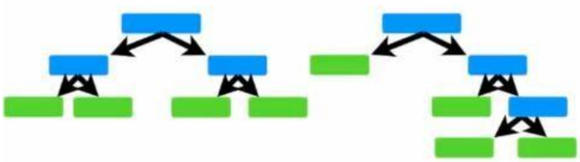
...so if we had to choose between using the **Random Forest** and **K-Nearest Neighbors**, we would choose the **Random Forest**.

	Has Heart Disease	Does Not Have Heart Disease
Has Heart Disease	142	22
Does Not Have Heart Disease	29	110

Lastly, we can apply **Logistic Regression** to the **Testing Dataset** and create a **Confusion Matrix**.

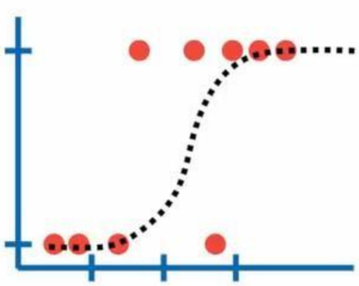


	Has Heart Disease	Does Not Have Heart Disease
Has Heart Disease	139	20
Does Not Have Heart Disease	32	112



	Has Heart Disease	Does Not Have Heart Disease
Has Heart Disease	142	22
Does Not Have Heart Disease	29	110

These two **Confusion Matrices** are very similar and make it hard to choose which machine learning method is a better fit for this data.



	Has Heart Disease	Does Not Have Heart Disease
Has Heart Disease	139	20
Does Not Have Heart Disease	32	112

Sensitivity, **Specificity**, **ROC** and **AUC**

		Actual		
		Troll 2	Gore Police	Cool as Ice
Predicted	Troll 2	12	102	93
	Gore Police	112	23	77
	Cool as Ice	83	92	17

...and if we had 40 things to choose from, we get a confusion matrix with 40 rows and 40 columns.

In summary, a **Confusion Matrix** tells you what your machine learning algorithm did right…

…and what it did wrong.

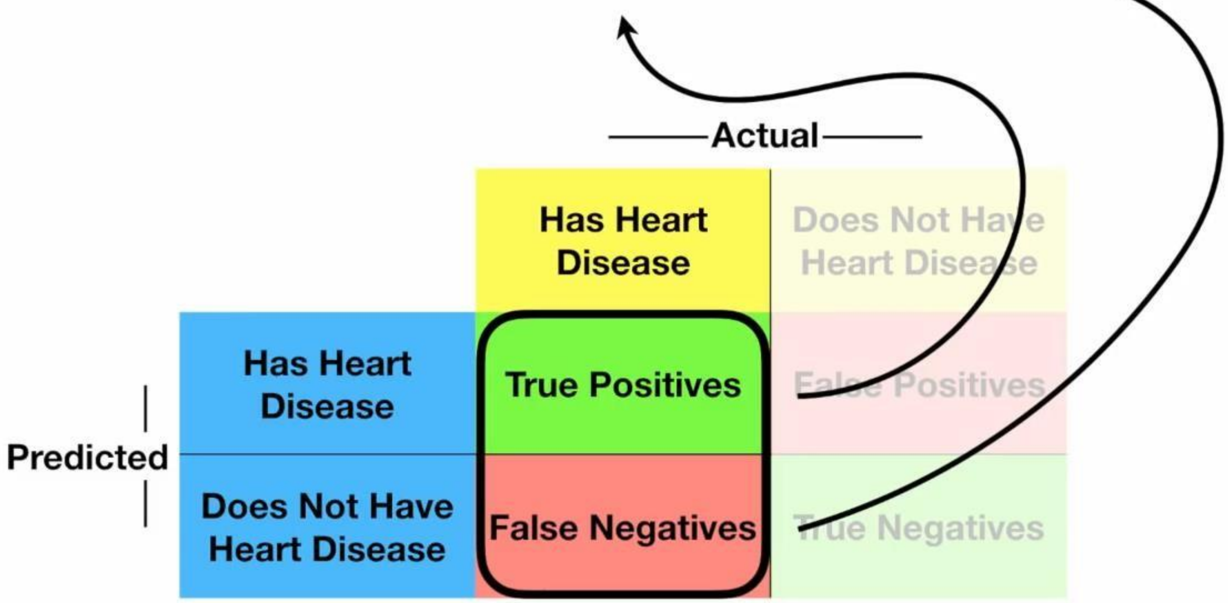
		Actual	
		Has Heart Disease	**Does Not Have Heart Disease**
Predicted	**Has Heart Disease**	**True Positives**	**False Positives**
	Does Not Have Heart Disease	**False Negatives**	**True Negatives**

Once we've filled out the **Confusion Matrix**, we can calculate two useful metrics: **Sensitivity** and **Specificity**.

		Actual	
		Has Heart Disease	**Does Not Have Heart Disease**
Predicted	**Has Heart Disease**	**True Positives**	**False Positives**
	Does Not Have Heart Disease	**False Negatives**	**True Negatives**

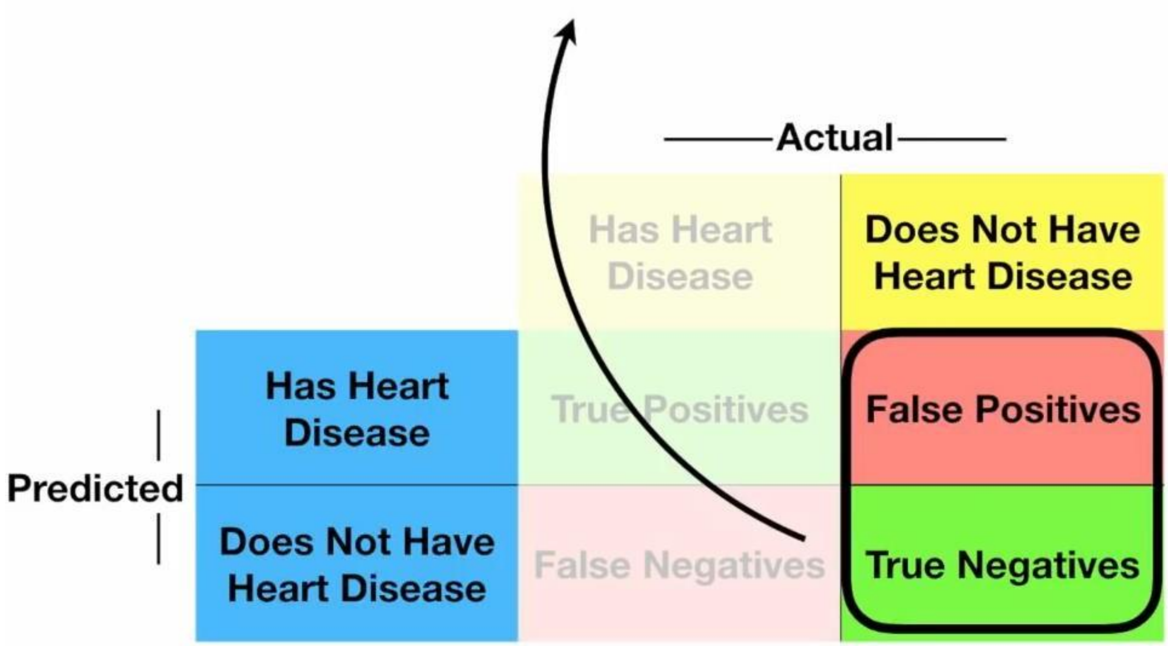
In this case, **Sensitivity** tells us what percentage of patients *with* heart disease were correctly identified.

$$\textbf{Sensitivity} = \frac{\text{True Positives}}{\text{True Positives} + \text{False Negatives}}$$

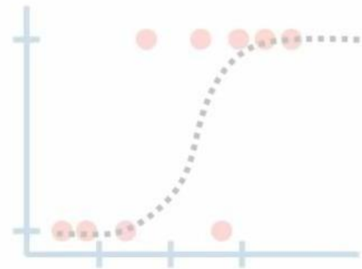


Specificity tells us what percentage of patients *without* heart disease were correctly identified.

$$\textbf{Specificity} = \frac{\text{True Negatives}}{\text{True Negatives} + \text{False Positives}}$$



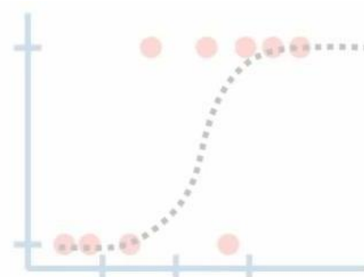
$$\textbf{Sensitivity} = \frac{139}{139 + 32} = 0.81$$



Sensitivity tells us that **81**% of the people *with* Heart Disease were correctly identified by the **Logistic Regression** model.

	Has Heart Disease	Does Not Have Heart Disease
Has Heart Disease	**139**	20
Does Not Have Heart Disease	**32**	112

$$\textbf{Specificity} = \frac{112}{112 + 20} = 0.85$$



Specificity tells us that **85**% of the people *without* Heart Disease were correctly identified by the **Logistic Regression** model.

	Has Heart Disease	Does Not Have Heart Disease
Has Heart Disease	139	**20**
Does Not Have Heart Disease	32	**112**

$$\textbf{Sensitivity} = \frac{142}{142 + 29} = 0.83$$

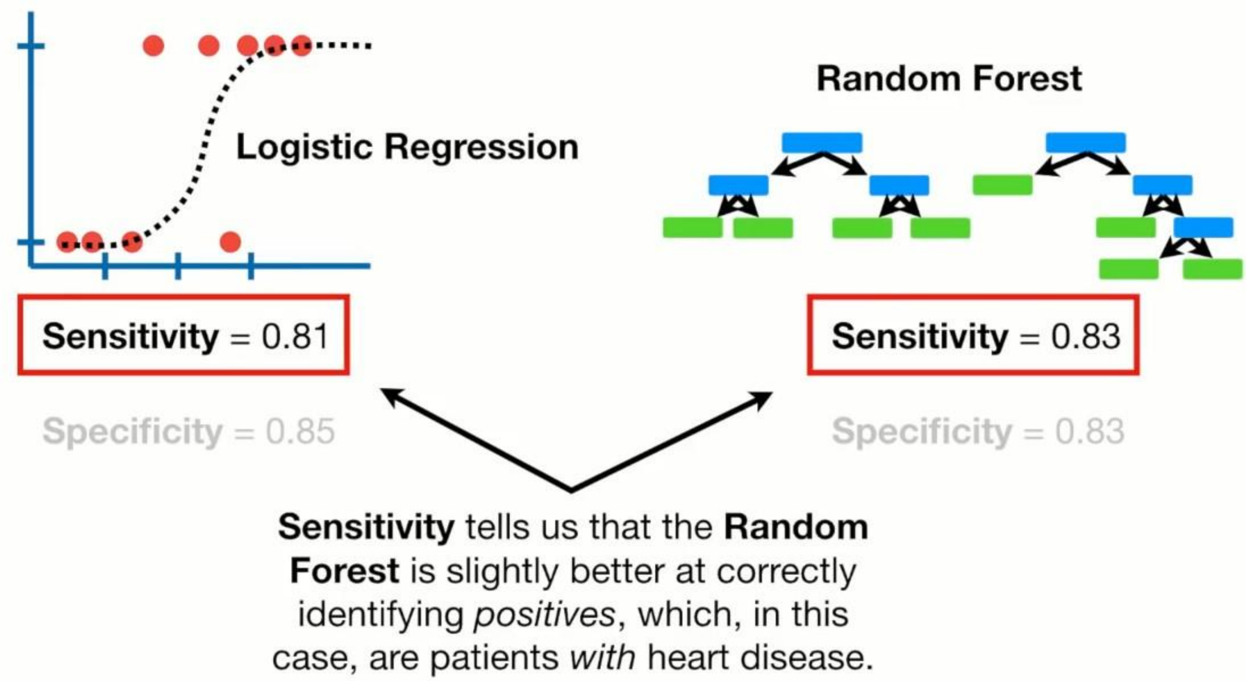
$$\textbf{Specificity} = \frac{\text{True Negatives}}{\text{True Negatives + False Positives}}$$

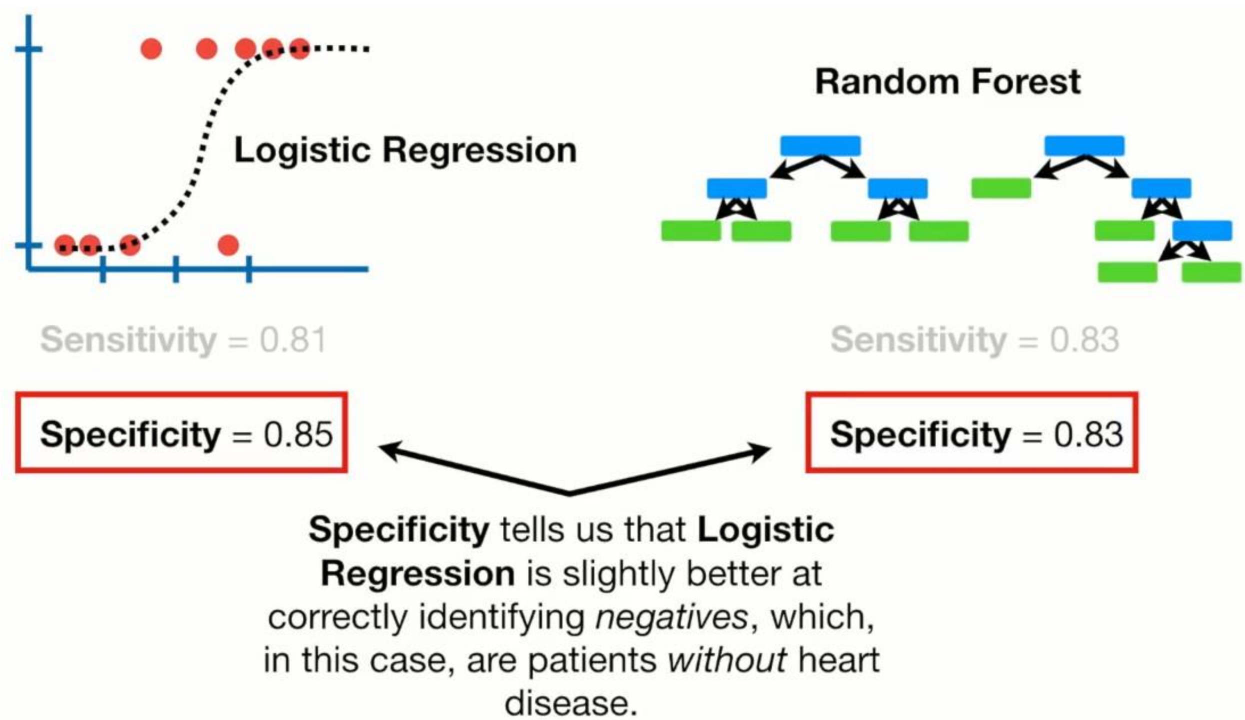
		Actual: Has Heart Disease	Actual: Does Not Have Heart Disease
Predicted	Has Heart Disease	142	22
Predicted	Does Not Have Heart Disease	29	110

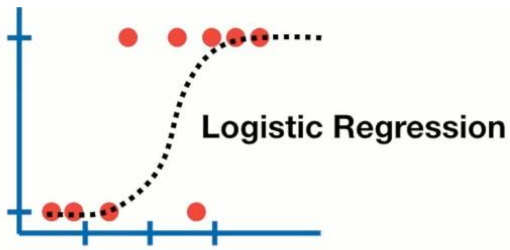
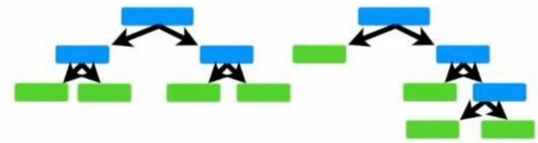
$$\textbf{Sensitivity} = \frac{142}{142 + 29} = 0.83$$

$$\textbf{Specificity} = \frac{110}{110 + 22} = 0.83$$

		Actual: Has Heart Disease	Actual: Does Not Have Heart Disease
Predicted	Has Heart Disease	142	22
Predicted	Does Not Have Heart Disease	29	110


Logistic Regression
Random Forest
Sensitivity = 0.81
Specificity = 0.85
Sensitivity = 0.83
Specificity = 0.83
Sensitivity tells us that the Random Forest is slightly better at correctly identifying positives, which, in this case, are patients with heart disease.


Logistic Regression
Random Forest
Sensitivity = 0.81
Specificity = 0.85
Sensitivity = 0.83
Specificity = 0.83
Specificity tells us that Logistic Regression is slightly better at correctly identifying negatives, which, in this case, are patients without heart disease.

 **Sensitivity** = 0.81 **Specificity** = 0.85 We would choose the **Logistic Regression** model if correctly identifying patients **without** heart disease was more important than correctly identifying patients **with** heart disease.	**Random Forest**  **Sensitivity** = 0.83 **Specificity** = 0.83 Alternatively, we would choose the **Random Forest** model if correctly identifying patients **with** heart disease was more important than correctly identifying patients **without** heart disease.

$$
\begin{aligned}
precision &= \frac{TP}{TP + FP} \\
recall &= \frac{TP}{TP + FN} \\
F1 &= \frac{2 \times precision \times recall}{precision + recall} \\
accuracy &= \frac{TP + TN}{TP + FN + TN + FP} \\
specificity &= \frac{TN}{TN + FP}
\end{aligned}
$$