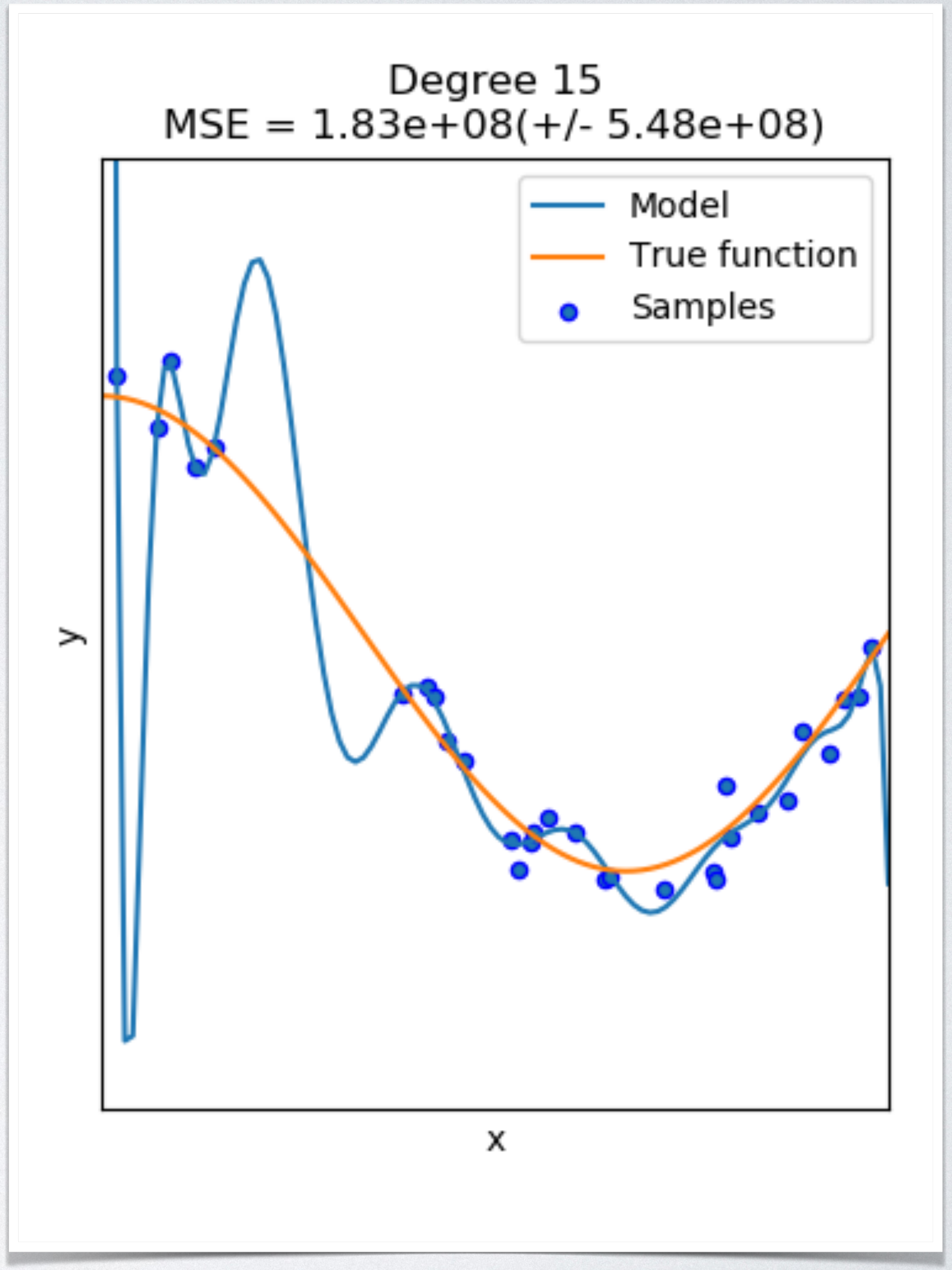
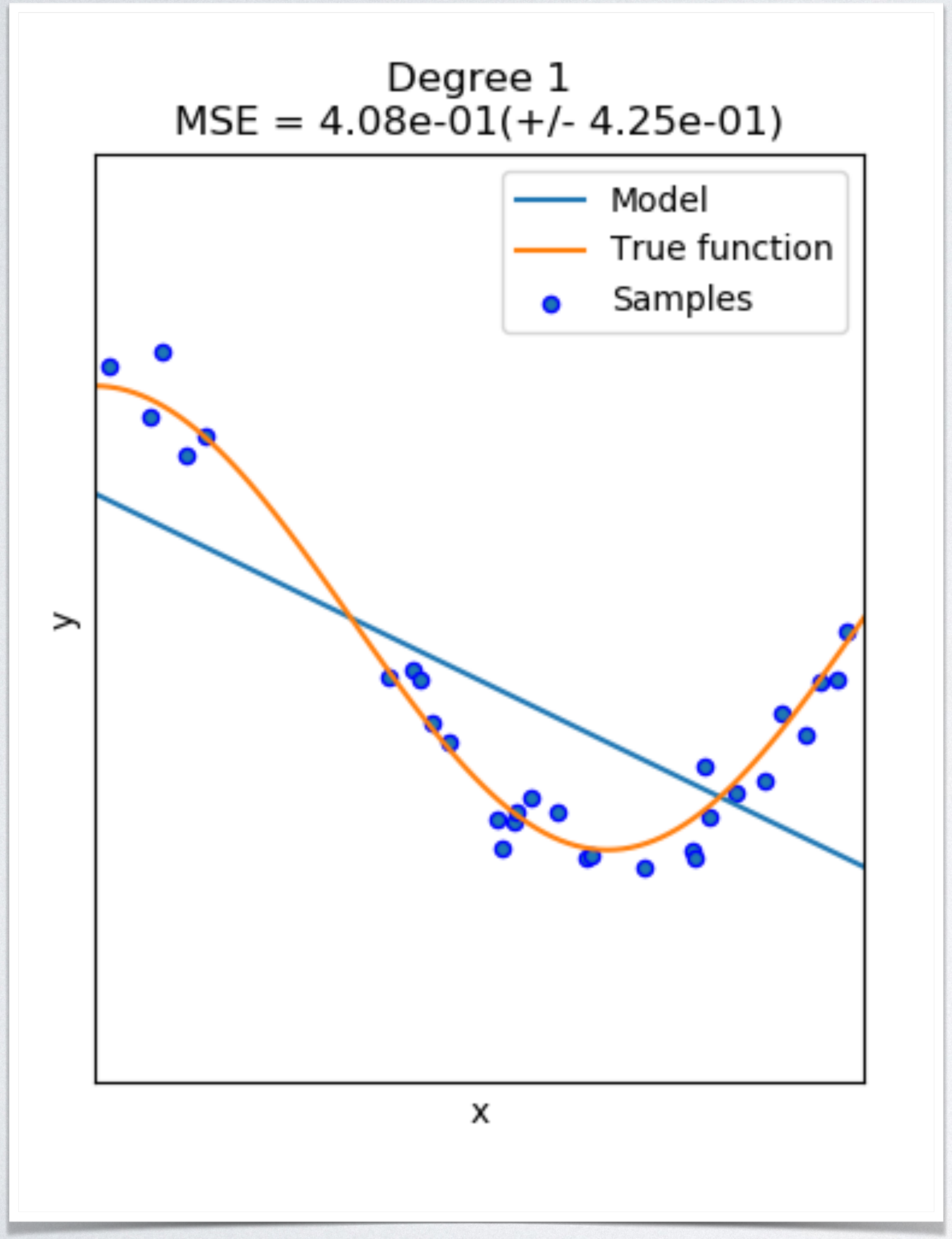
MODEL EVALUATION AND SELECTION

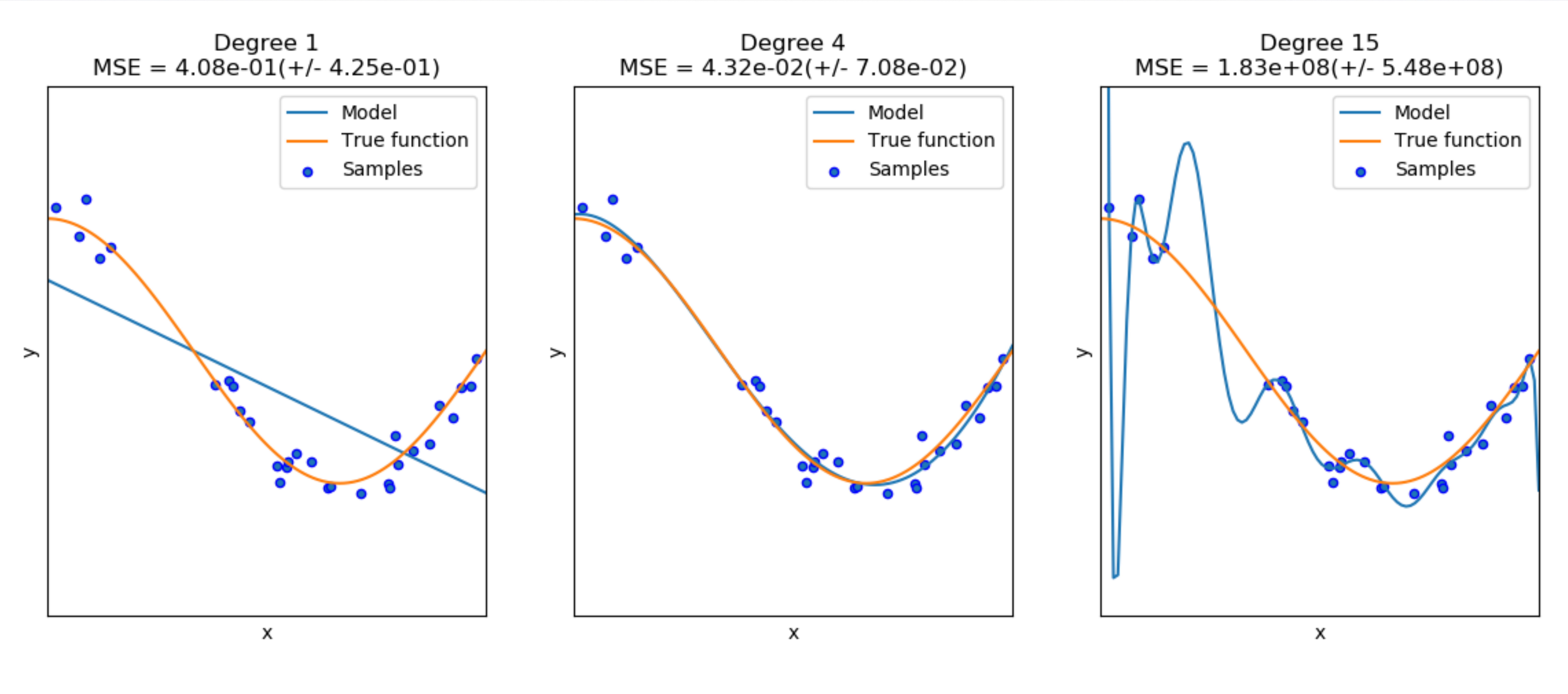
PERFECT MATCH TO YOUR DATA! OR...

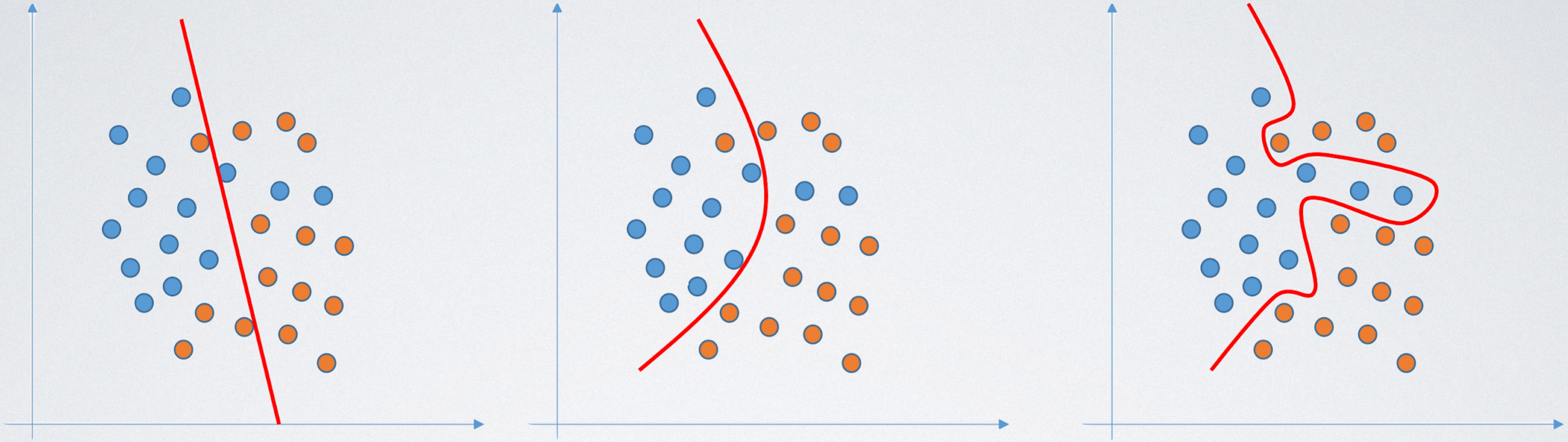


PERFECT MATCH TO YOUR DATA! OR...



PERFECT MATCH TO YOUR DATA! OR...





underfitting

overfitting

GENERAL STRATEGY

Training data

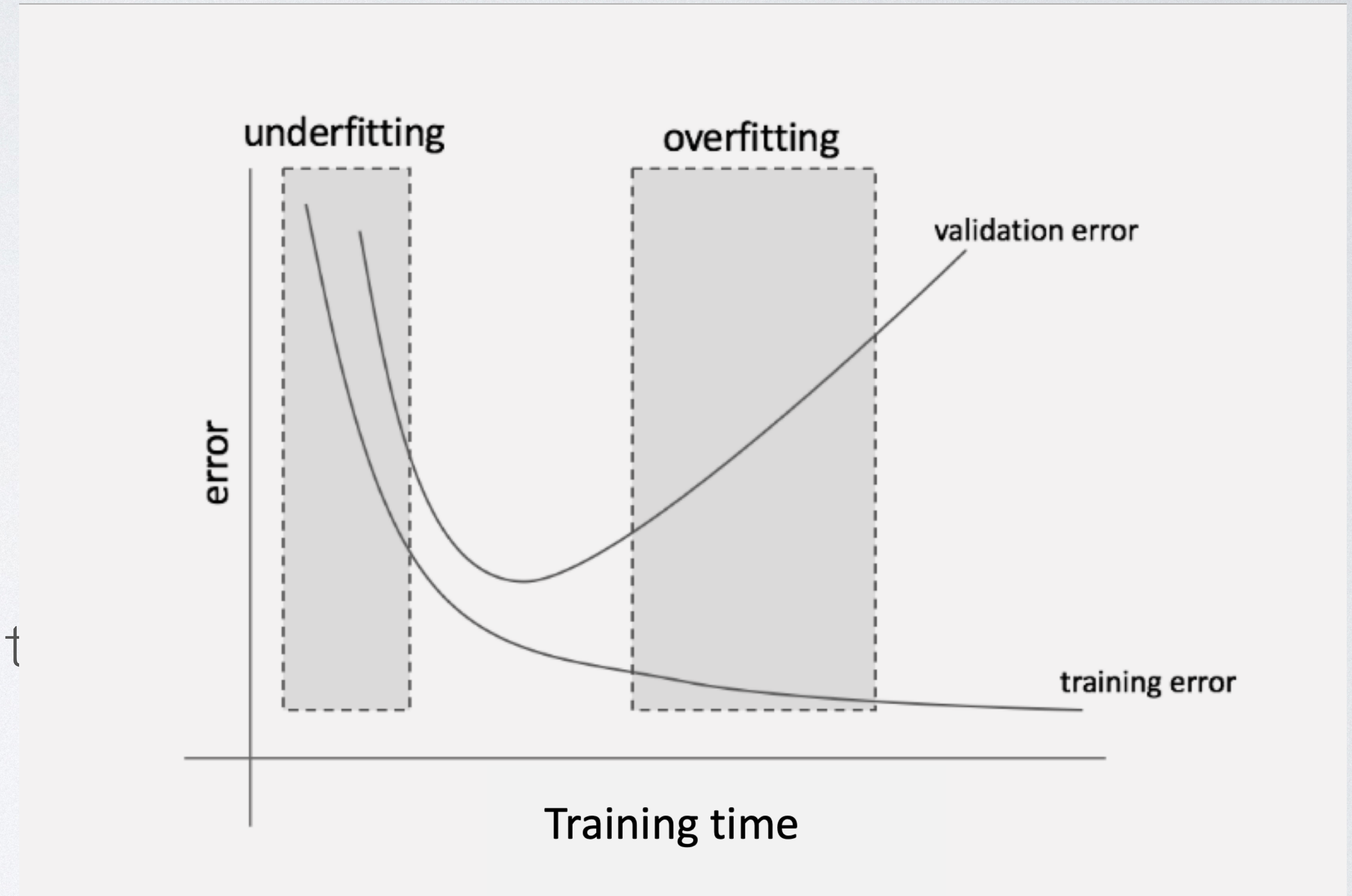
- Used to tune (train) the algorithm

Validation data

- Used to choose best algorithm or to find t

Test data

- Used to evaluate the accuracy of the model



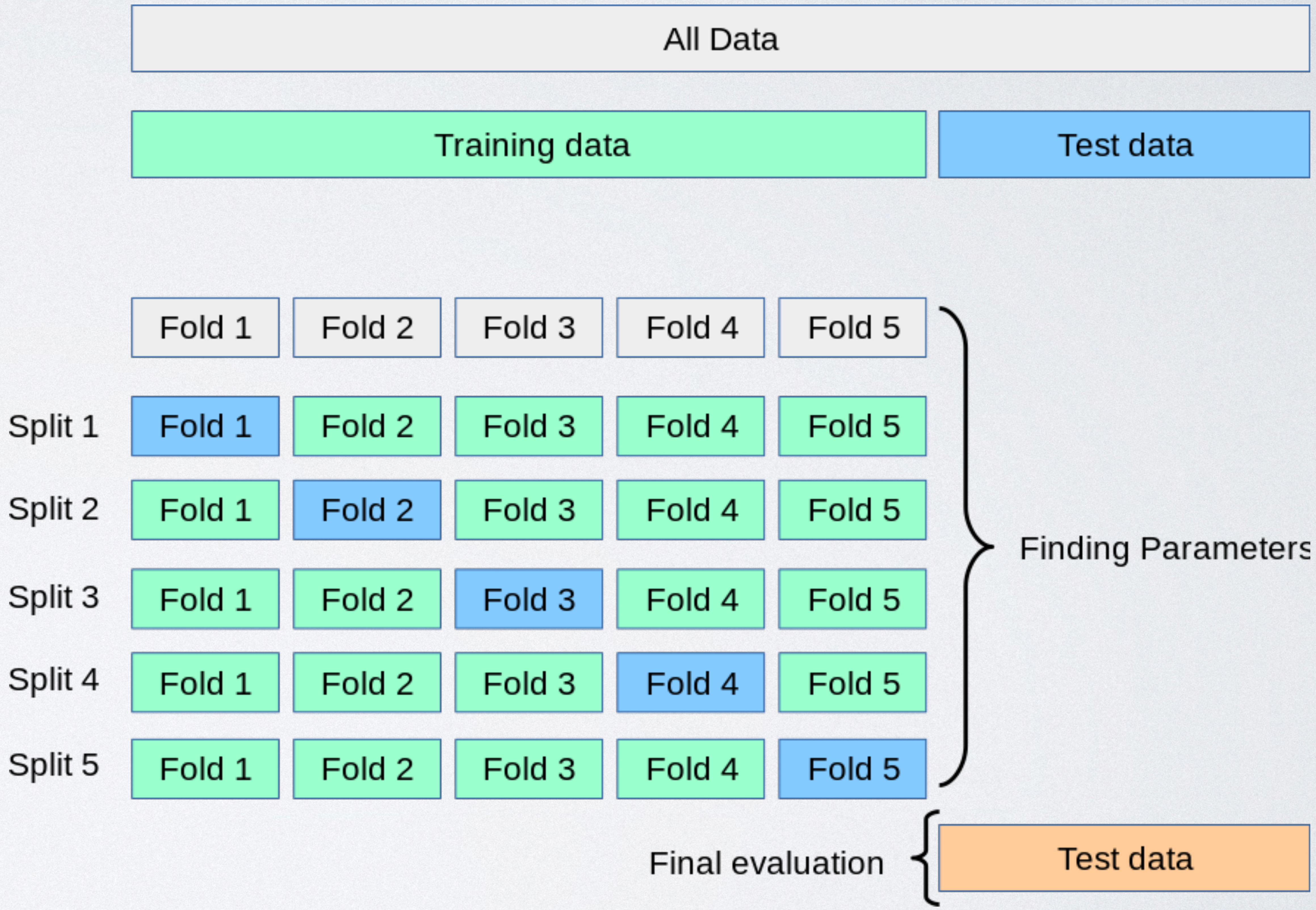
BUT I WANT TO USE ALL MY DATA FOR TRAINING!

K-FOLD CROSS VALIDATION

Leave-one-out

Stratified cross-validation
(class label distribution similar in each fold)

Stratified 10-fold cross-validation



https://scikit-learn.org/stable/modules/cross_validation.html

BREAK

CLASSIFICATION ERRORS

- Will just consider binary classification
- Can be extended to multiclass problems

CONFUSION MATRIX

		Predicted class		
		Positive	Negative	Total
Actual class	Positive	*TP*	*FN*	*P*
	Negative	*FP*	*TN*	*N*
	Total	*P'*	*N'*	*P + N*

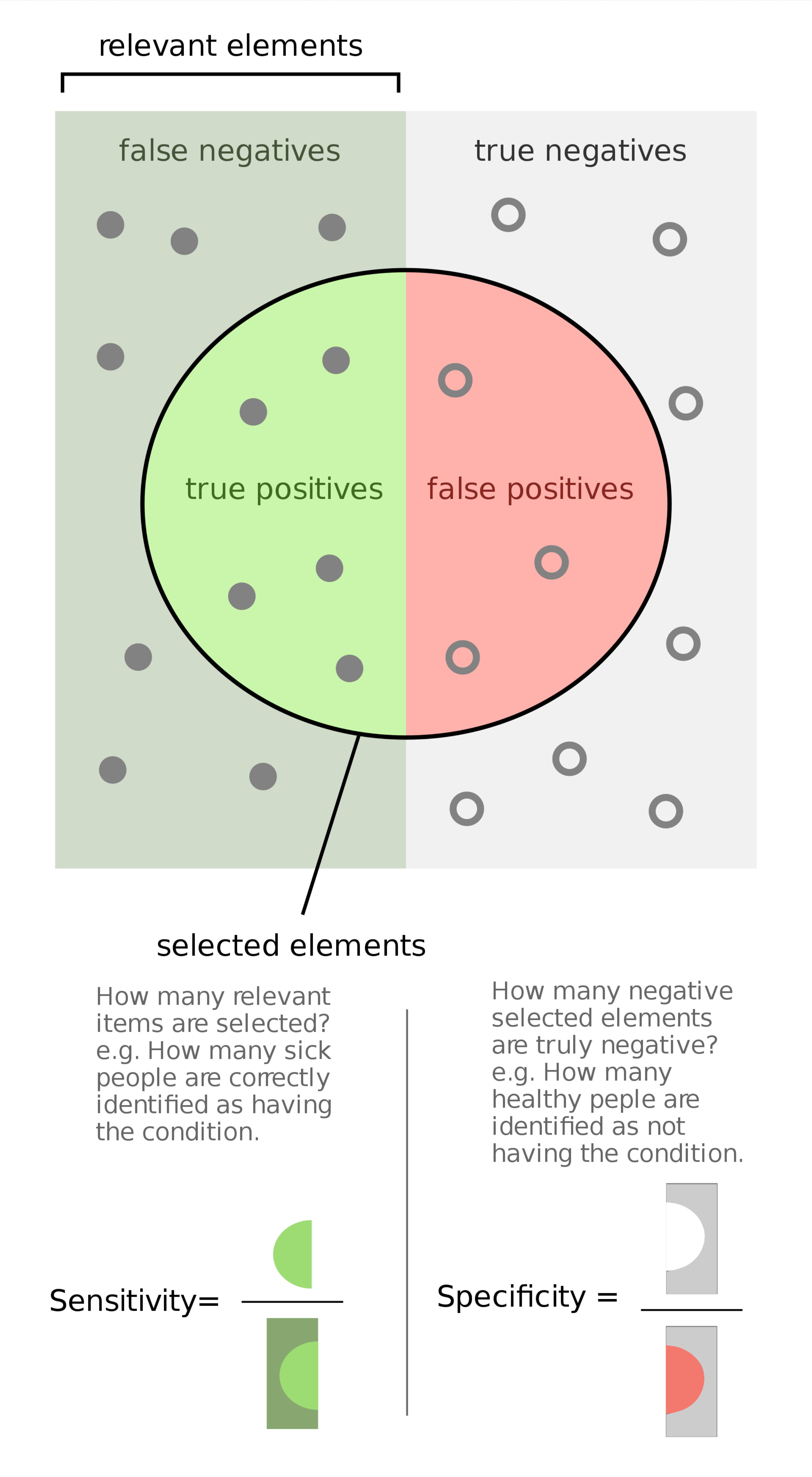
EVALUATION MEASURES

		Predicted Class: Positive	Predicted Class: Negative	
Actual Class	Positive	True Positive (TP)	False Negative (FN) **Type II Error**	**Sensitivity** $\frac{TP}{(TP+FN)}$
Actual Class	Negative	False Positive (FP) **Type I Error**	True Negative (TN)	**Specificity** $\frac{TN}{(TN+FP)}$
		Precision $\frac{TP}{(TP+FP)}$	**Negative Predictive Value** $\frac{TN}{(TN+FN)}$	**Accuracy** $\frac{TP+TN}{(TP+TN+FP+FN)}$

"guilty person not convicted"

"Innocent person convicted"

https://manisha-sirsat.blogspot.com/2019/04/confusion-matrix.html



FeanDoe - Modified version from Walber's Precision and Recall
https://commons.wikimedia.org/wiki/File:Precisionrecall.svg

ACCURACY

Will a given user make a purchase in the my game?

$(TP + TN) / (N + P)$

Can you think of a model that has >95% accuracy?

ACCURACY

- Will a given user make a purchase in my game?
- Accuracy = (TP + TN) / (N + P)

(0 + 97) / 100 = 97%

	Predicted class			
Actual		Positive	Negative	Total
	Positive	0	3	3
	Negative	0	97	97
	Total	0	100	100

EXERCISE BREAK

Calculate the other error measurements

		Predicted class		
		Positive	Negative	Total
Actual	Positive	0	3	3
	Negative	0	97	97
	Total	0	100	100

OTHER MEASURES

Sensitivity/recall: TP / P = **0%**

Specificity: TN / N = 97 / 97 = **100%**

Precision = TP / (TP + FP) = **0%**

Overall a pretty horrible model

Accuracy may be misleading with big class imbalances!

		Predicted class		
		Positive	Negative	Total
Actual	Positive	0	3	3
	Negative	0	97	97
	Total	0	100	100

SLIGHTLY MORE REALISTIC

Sensitivity/recall: *TP* / *P* = 2 / 3 = **67%**

Specificity: *TN* / *N* = 95 / 97 = **98%**

Precision = *TP* / (*TP* + *FP*) = 2 / (2 + 2) = **50%**

If it's important to find **all positives**: high sensitivity/recall

If it's important to **only target positives**: high precision

		Predicted class		
		Positive	Negative	Total
Actual	Positive	2	1	3
	Negative	2	95	97
	Total	4	96	100

F-SCORES

F: harmonic mean of precision and recall
Closer to truest average with rates/ratios
I.e. NOT (*precision* + *recall*) / 2

$$F = \frac{2 \times precision \times recall}{precision + recall}$$

β: more weight to recall than precision
Dependent on cost of misclassification

$$F_\beta = \frac{(1 + \beta^2) \times precision \times recall}{\beta^2 \times precision + recall}$$