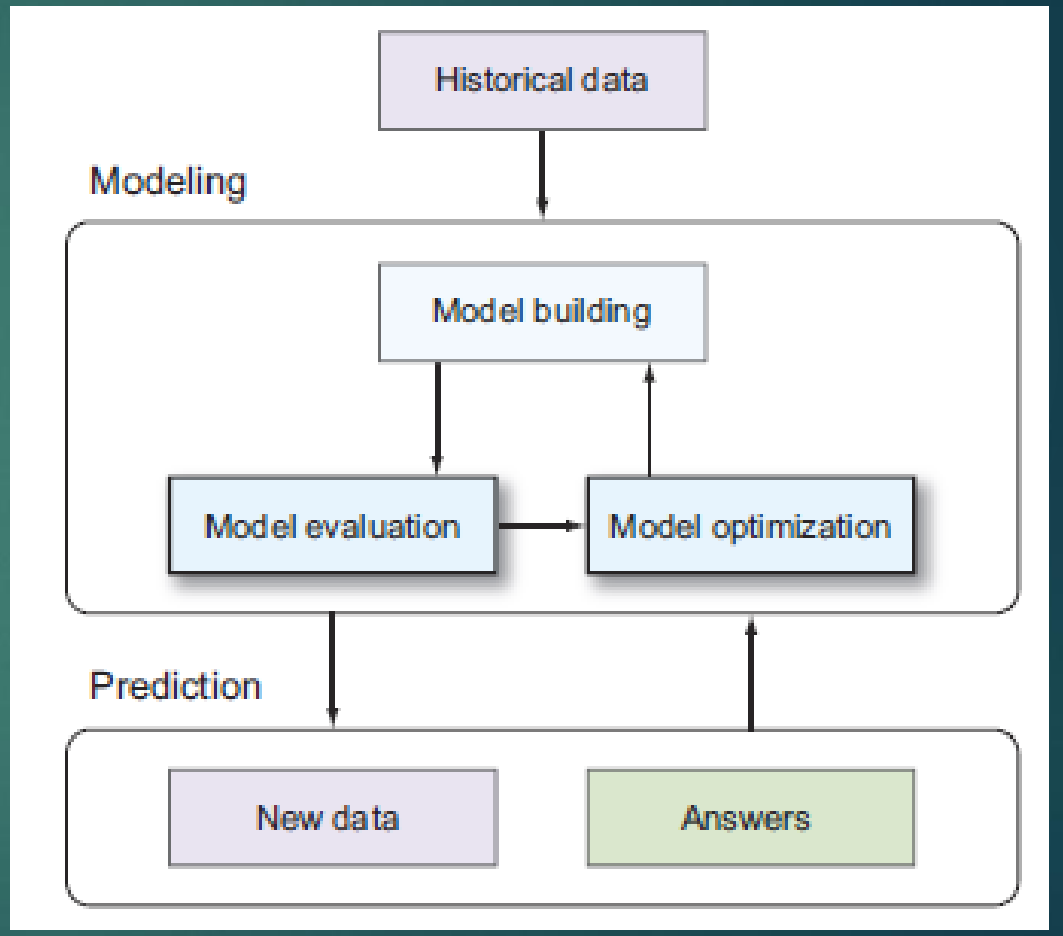
Applied Machine Learning: Model Optimization

NAWWAF KHARMA

Model Evaluation and Optimization

- By the end of this chapter, you'll be equipped with the means and know-how to evaluate the predictive accuracy of the ML models that you built in the previous chapter
- 4.1 Model generalization: assessing predictive accuracy for new data

Therefore, when you evaluate the performance of a model, you want to determine how well that model will perform on new data.



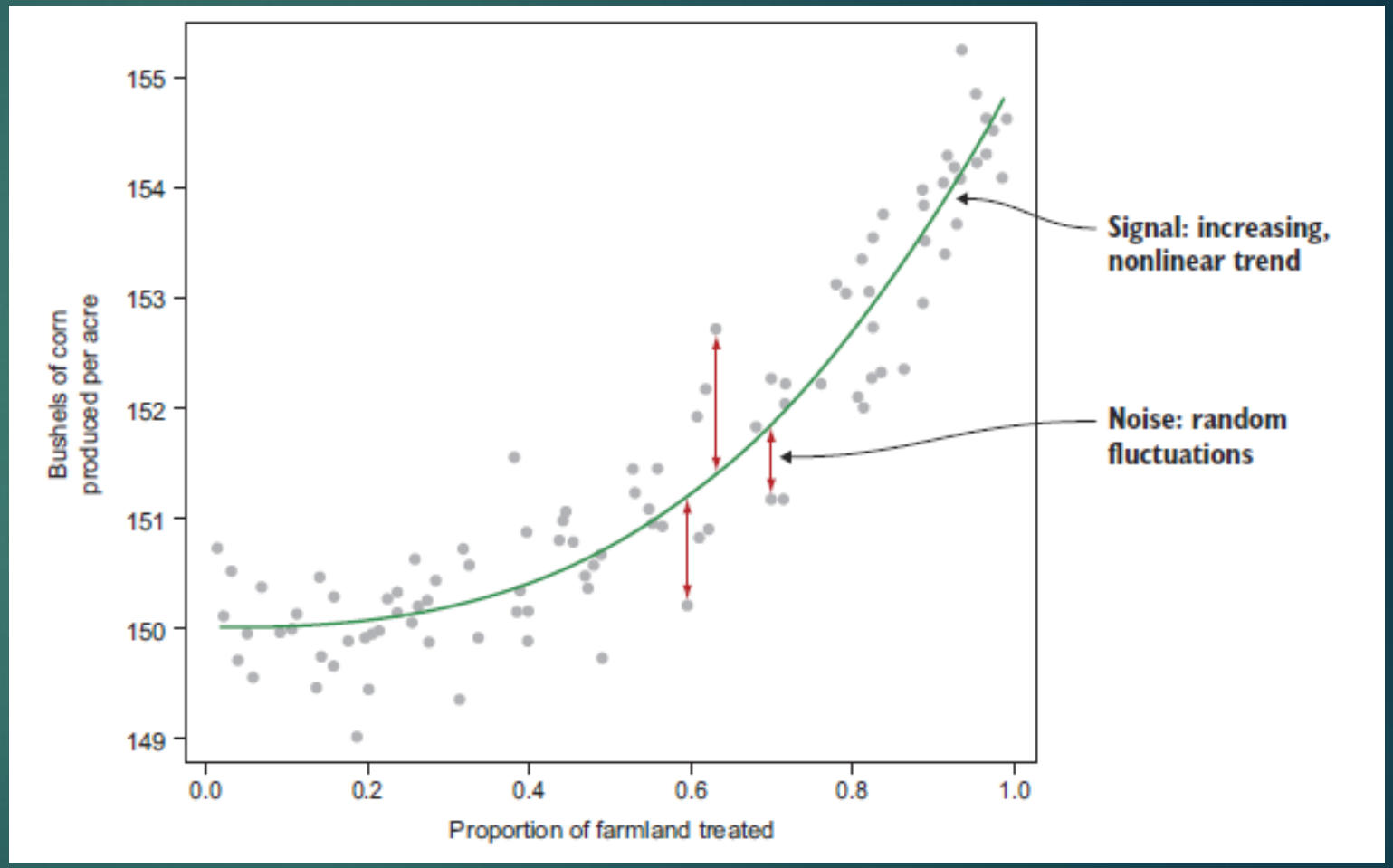
Model Evaluation and Optimization

- 4.1.1 The problem: overfitting and model optimism

One of the simplest ML regression models is kernel smoothing. Kernel smoothing operates by taking local averages:

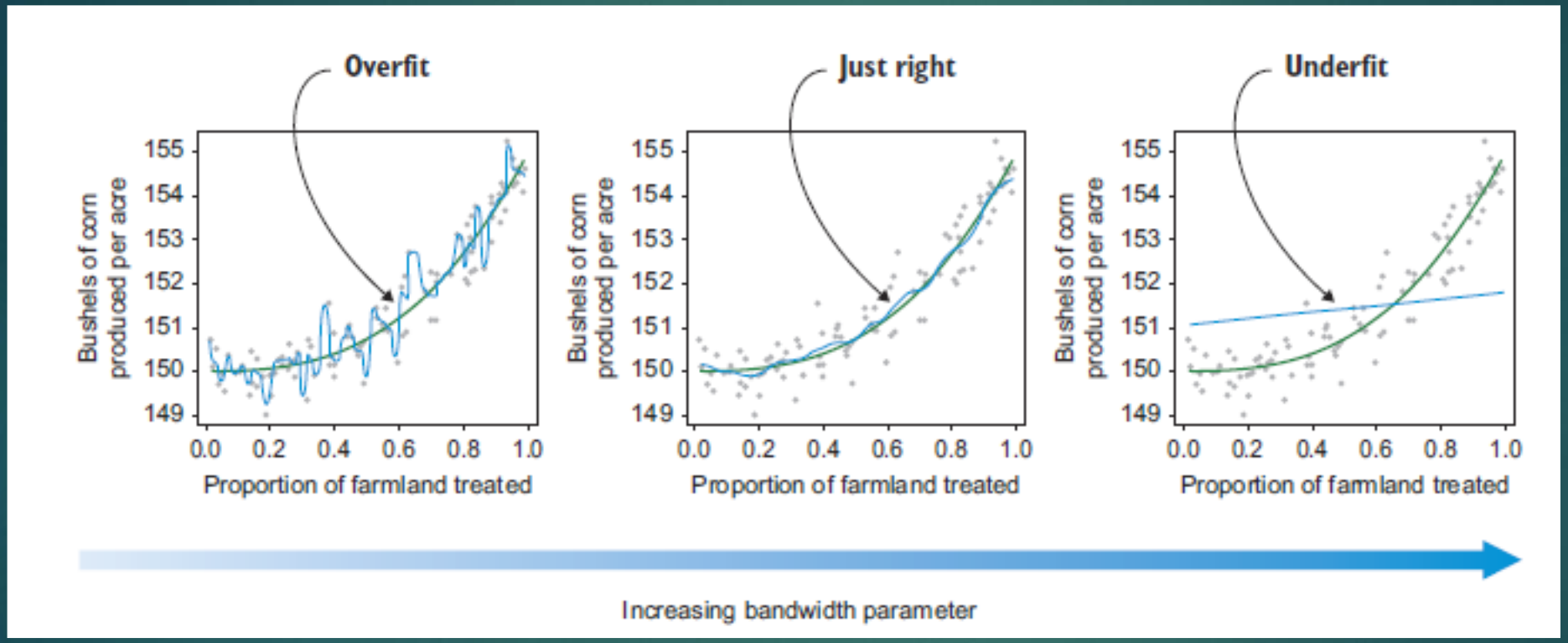
for each new data point, the value of the target variable is modeled as the *average* of the target variable for only the training data whose feature value is **close to the feature value** of the new data point.

A single parameter, called the **bandwidth parameter**, controls the size of the window for the local averaging.



Model Evaluation and Optimization

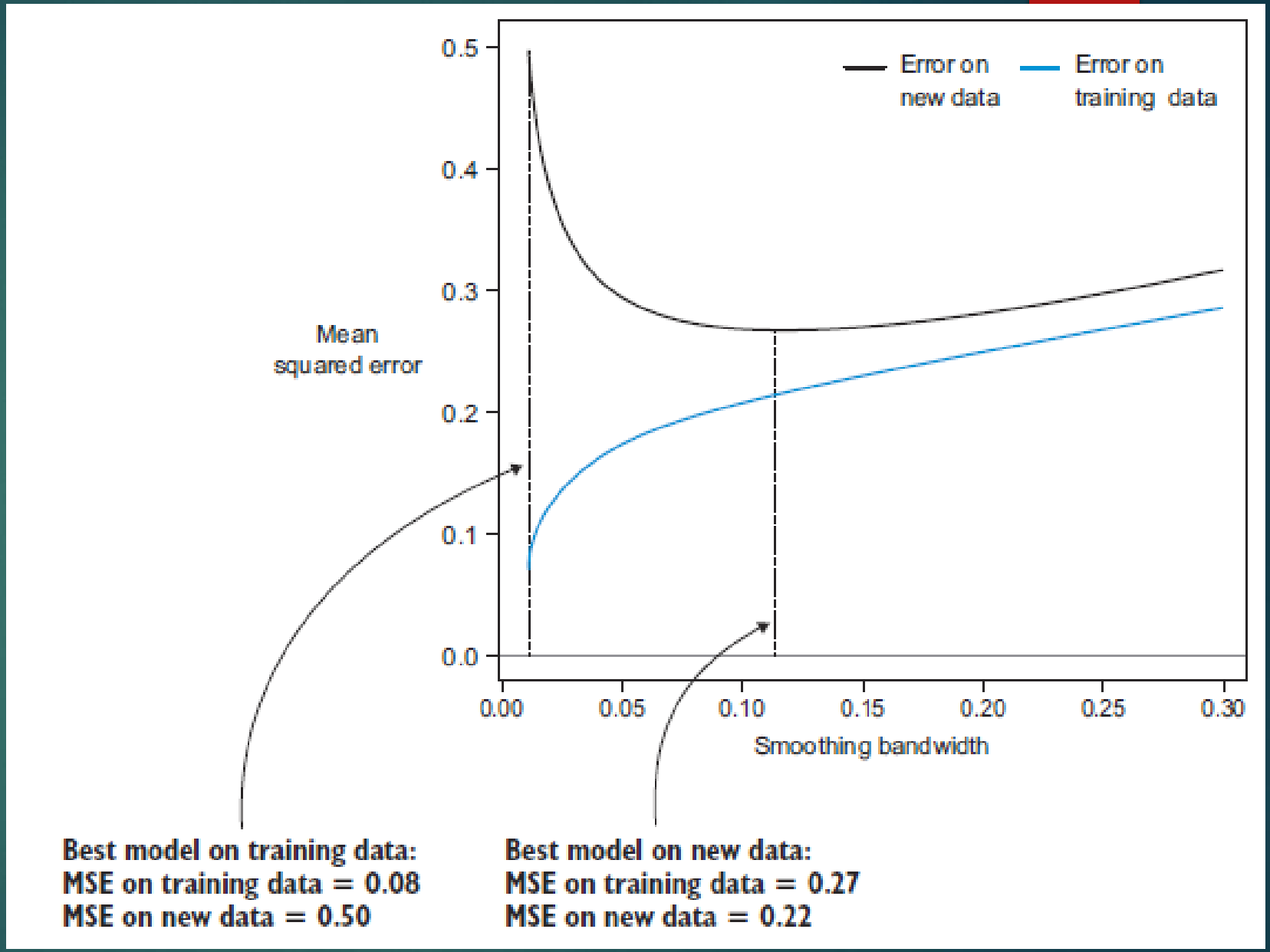
- various values of the kernel-smoothing bandwidth parameters: results



- For regression, the standard metric for evaluation is mean squared error (MSE), which is the average of the squared differences between the true value of the target variable and the model-predicted value

Cont.

- Simply put, the performance of the predictions of a model evaluated on the training set isn't indicative of the performance of that model on new data.
- CAUTION ABOUT DOUBLE-DIPPING THE TRAINING DATA Using the training data for both model fitting and evaluation purposes can lead you to be overly optimistic



Cont.

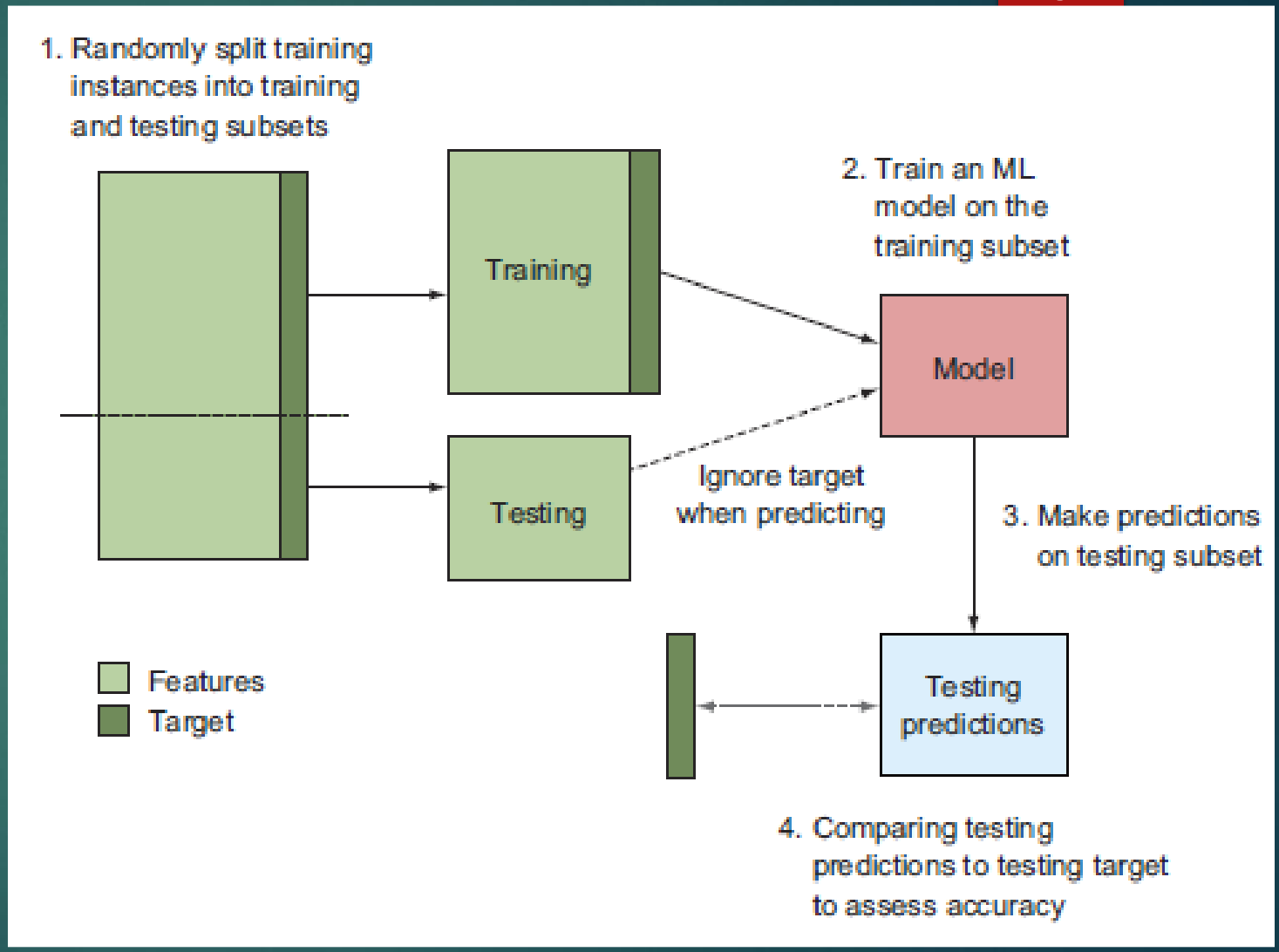
You need an evaluation metric that better approximates the performance of the model on new data.

4.1.2 The solution: **cross-validation**

(A) THE HOLDOUT METHOD

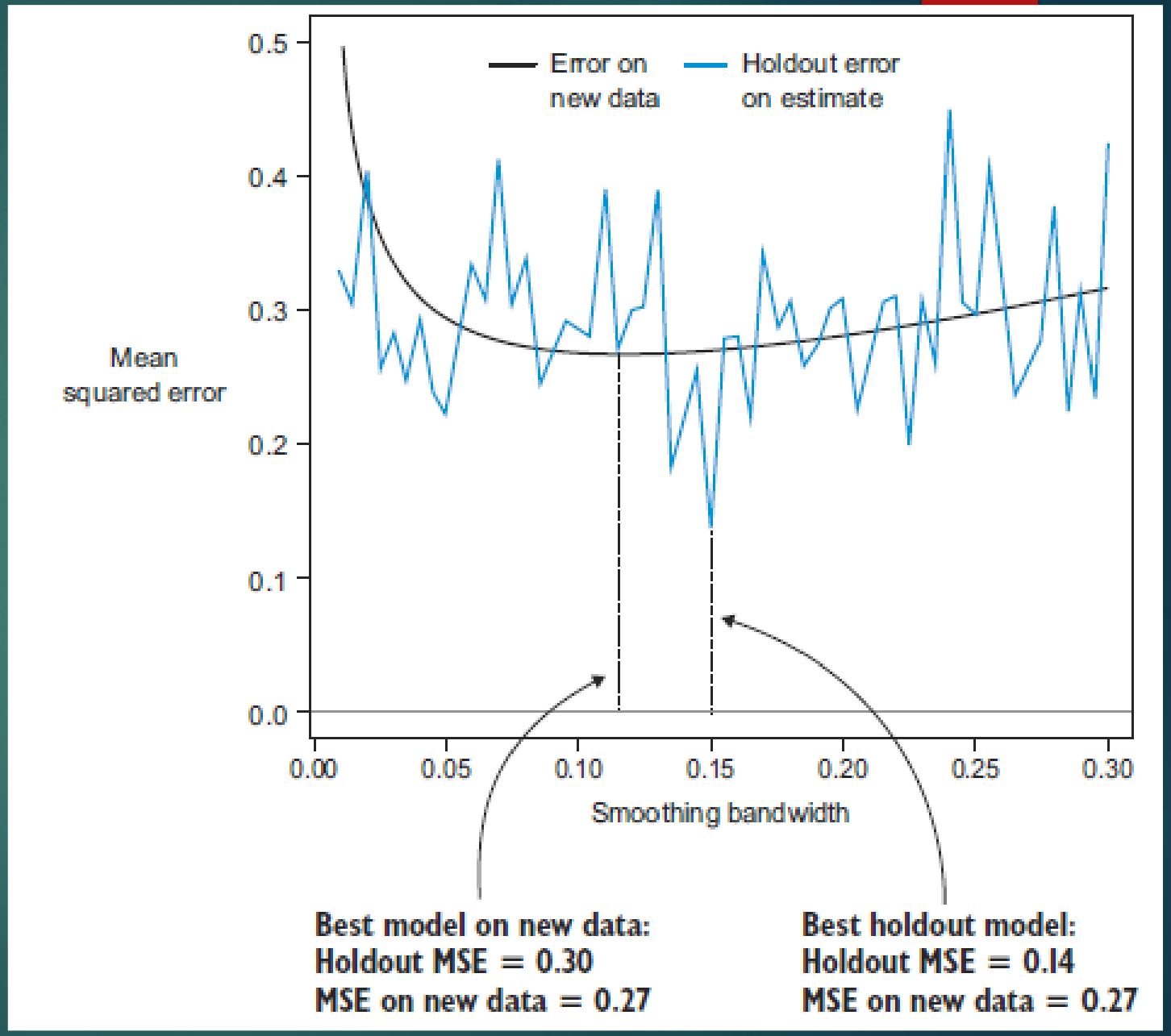
* The error estimates computed by the holdout method are close to the new-data error of the model.

* The holdout error estimates are noisy.



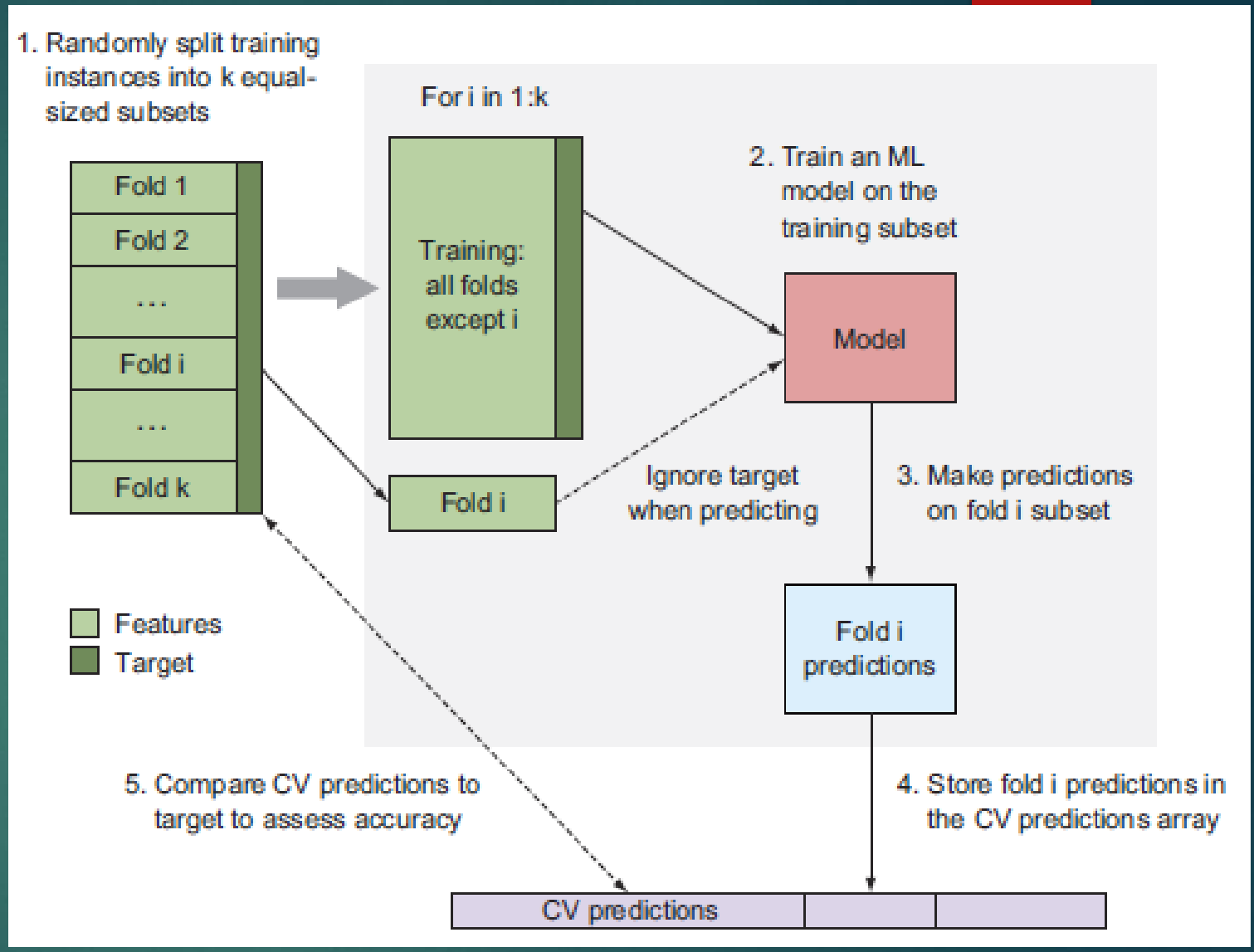
Cont.

- You could beat down the noise by doing repeated random training-testing splits and averaging the result. But over multiple iterations, each data point will be assigned to the testing set a different number of times, which could bias the result.
- A better solution is k-fold cross-validation ..



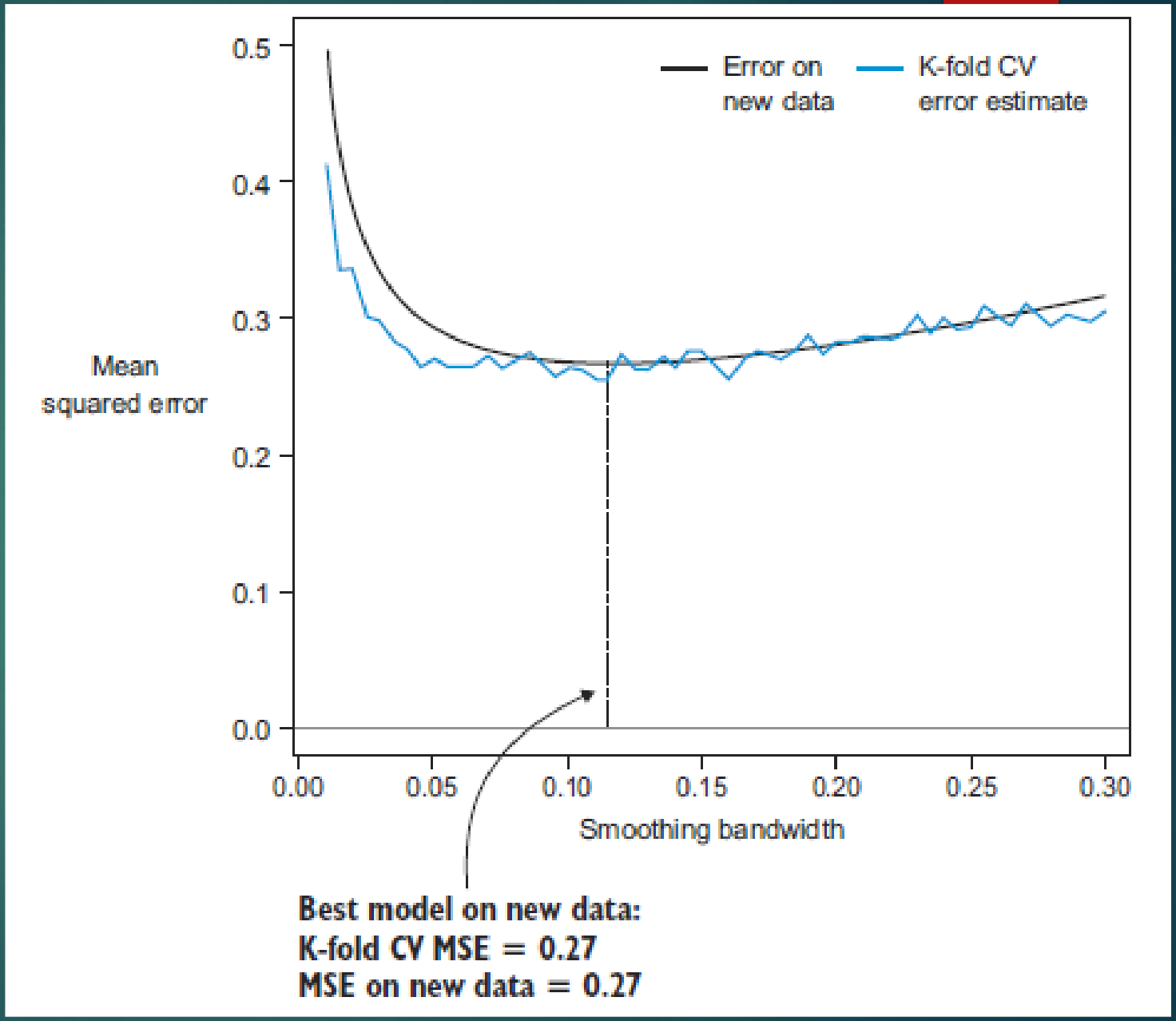
Cont.

- Algorithm: ..



Cont.

- Comparison of the k-fold cross-validation error MSE to the MSE on new data, using the corn production dataset.
- The k-fold CV error is a good estimate for how the model will perform on new data ..
- allowing you to use it confidently to forecast the error of the model and to select the best model.



Cont.

- 4.1.3 Some things to look out for when using cross-validation

1. Cross-validation methods (including both the holdout and k-fold methods) assume that the training data forms a **representative** sample from the population of interest. A sufficiently large sample, randomly selected from the whole population; for size determination, see: https://www.wikihow.com/Calculate-Sample-Size
2. Some datasets use features that are **temporal**-- If this is the case with your data, you must ensure that features that are available in the future can *never* be used to predict the past. I.e., all training set instances come *before* test set instances.
3. The larger the number of folds used in k-fold cross-validation, the *better* the error estimates will be, but the *longer* your program will take to run. Solution: Use at least 10 folds

Cont.

- 4.2 Evaluation of classification models: binary classification

Imagine that you want to predict whether a Titanic passenger would survive, based on personal, social, and economic factors

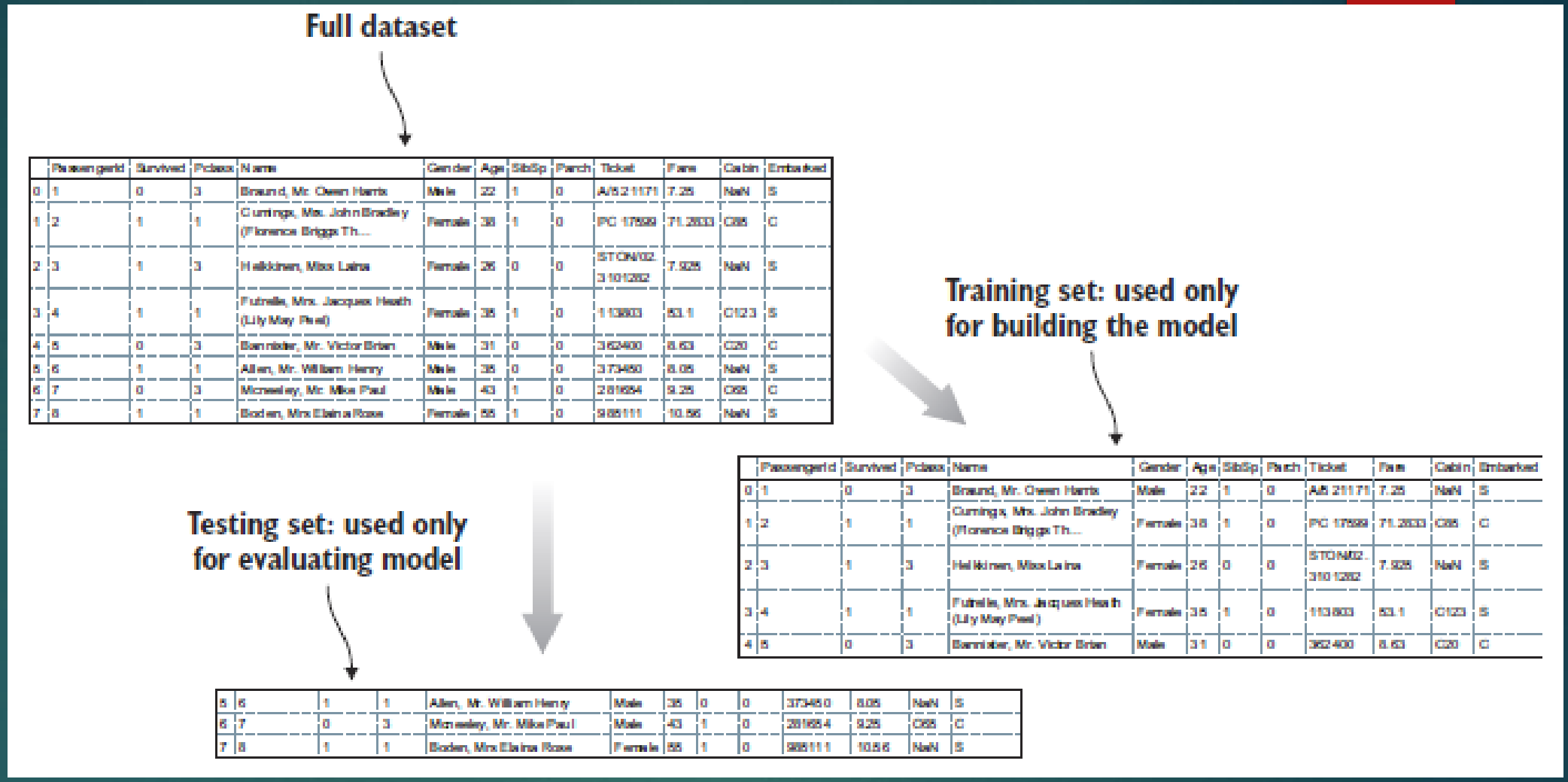
The goal of this section is to evaluate the model in order to optimize the prediction accuracy and compare with other models.

Target column

	PassengerId	Survived	Pclass	Name	Gender	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	Male	22	1	0	A/5 21171	7.25	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th…	Female	38	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss Laina	Female	26	0	0	STON/02. 3101282	7.925	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	Female	35	1	0	113803	53.1	C123	S
4	5	0	3	Allen, Mr. William Henry	Male	35	0	0	373450	8.05	NaN	S

Cont.

- Splitting the full dataset into training and testing sets allows you to evaluate the model.



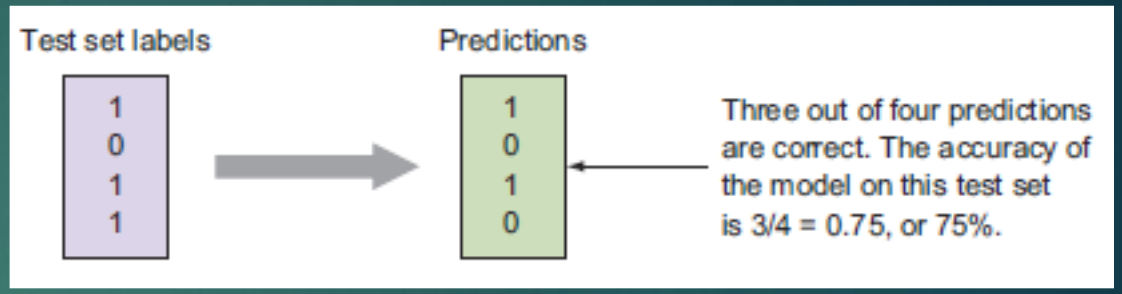
	PassengerId	Survived	Pclass	Name	Gender	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	Male	22	1	0	A/5 21171	7.25	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	Female	38	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss Laina	Female	26	0	0	STON/O2. 3101282	7.925	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	Female	35	1	0	113803	53.1	C123	S
4	5	0	3	Bannister, Mr. Victor Brian	Male	31	0	0	362400	8.63	C20	C
5	6	1	1	Allen, Mr. William Henry	Male	35	0	0	373450	8.05	NaN	S
6	7	0	3	Mcnesley, Mr. Mike Paul	Male	43	1	0	2251654	9.25	C65	C
7	8	1	1	Boden, Mrs Elaina Rose	Female	55	1	0	9851111	10.56	NaN	S

	PassengerId	Survived	Pclass	Name	Gender	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
0	1	0	3	Braund, Mr. Owen Harris	Male	22	1	0	A/5 21171	7.25	NaN	S
1	2	1	1	Cumings, Mrs. John Bradley (Florence Briggs Th...	Female	38	1	0	PC 17599	71.2833	C85	C
2	3	1	3	Heikkinen, Miss Laina	Female	26	0	0	STON/O2. 3101282	7.925	NaN	S
3	4	1	1	Futrelle, Mrs. Jacques Heath (Lily May Peel)	Female	35	1	0	113803	53.1	C123	S
4	5	0	3	Bannister, Mr. Victor Brian	Male	31	0	0	362400	8.63	C20	C

5	6	1	1	Allen, Mr. William Henry	Male	35	0	0	373450	8.05	NaN	S
6	7	0	3	Mcnesley, Mr. Mike Paul	Male	43	1	0	2251654	9.25	C65	C
7	8	1	1	Boden, Mrs Elaina Rose	Female	55	1	0	9851111	10.56	NaN	S

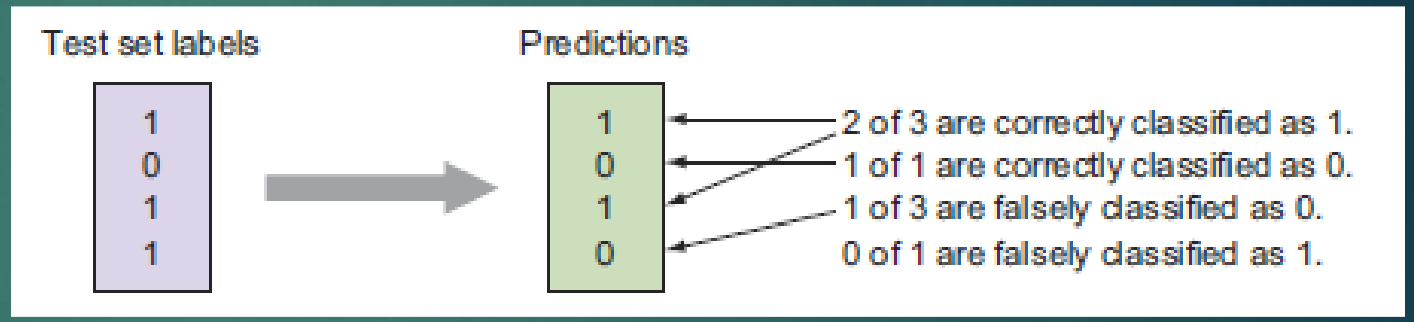
Cont.

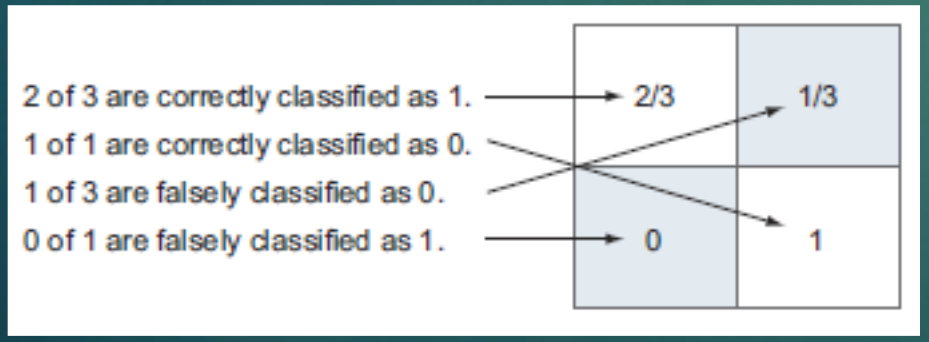
- The simplest performance measure of a classification model is to calculate the fraction of correct answers



- 4.2.1 Class-wise accuracy

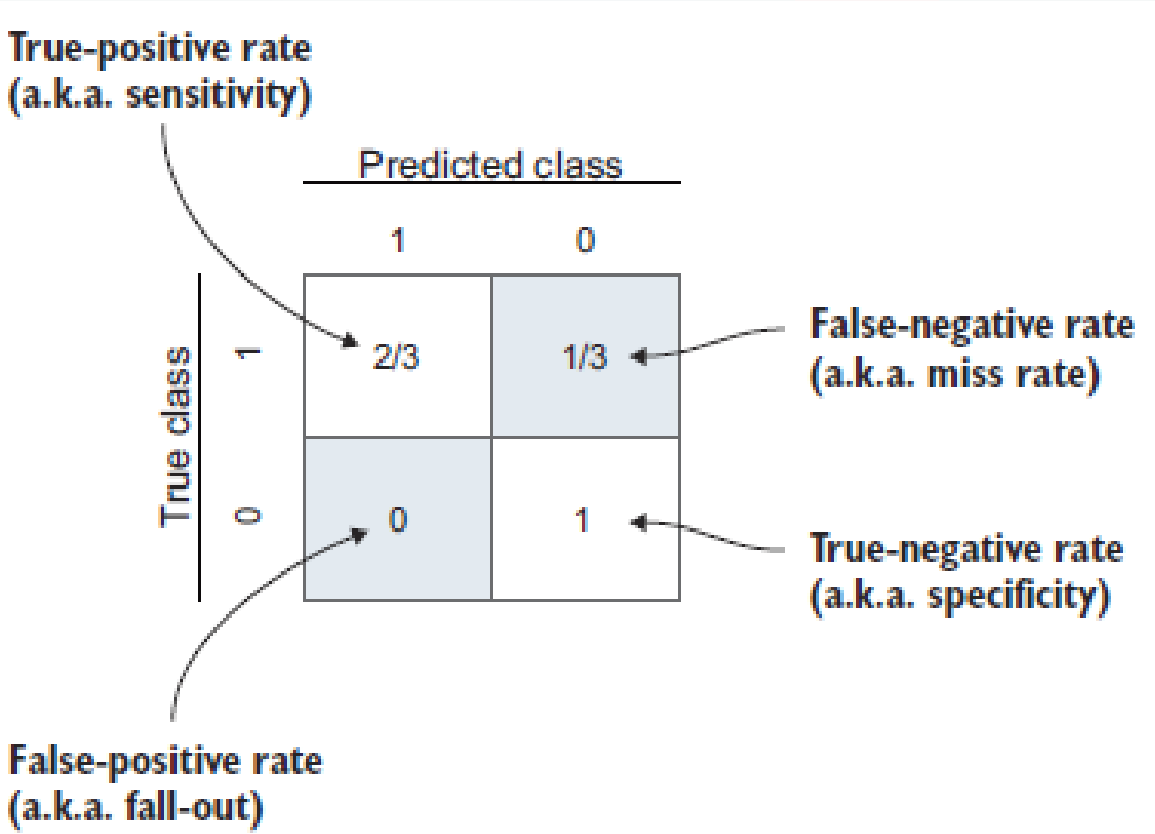
and the **confusion matrix**

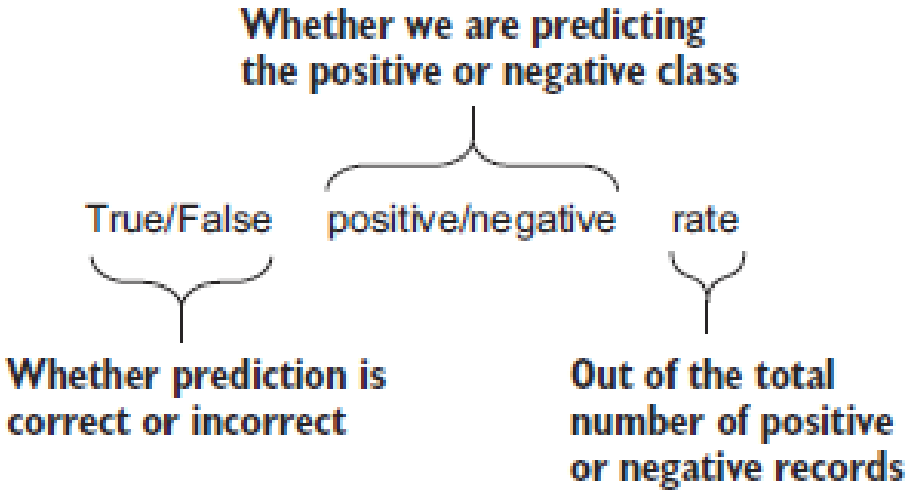




Cont.

Figure here slide relates the specific confusion matrix in previous slide to the general concept of receiver operating characteristics (ROCs) that you'll employ widely throughout the rest of this book.





Cont.

- 4.2.2 Accuracy trade-offs and ROC curves

Many classification algorithms output not only the zero-one predictions, but the full prediction probabilities (or equivalent)

Output from classifier: class probabilities

	Survived	Died
15	0.092	0.908
16	0.904	0.096
17	0.646	0.354
18	0.740	0.260
19	0.460	0.540

Sorted probabilities

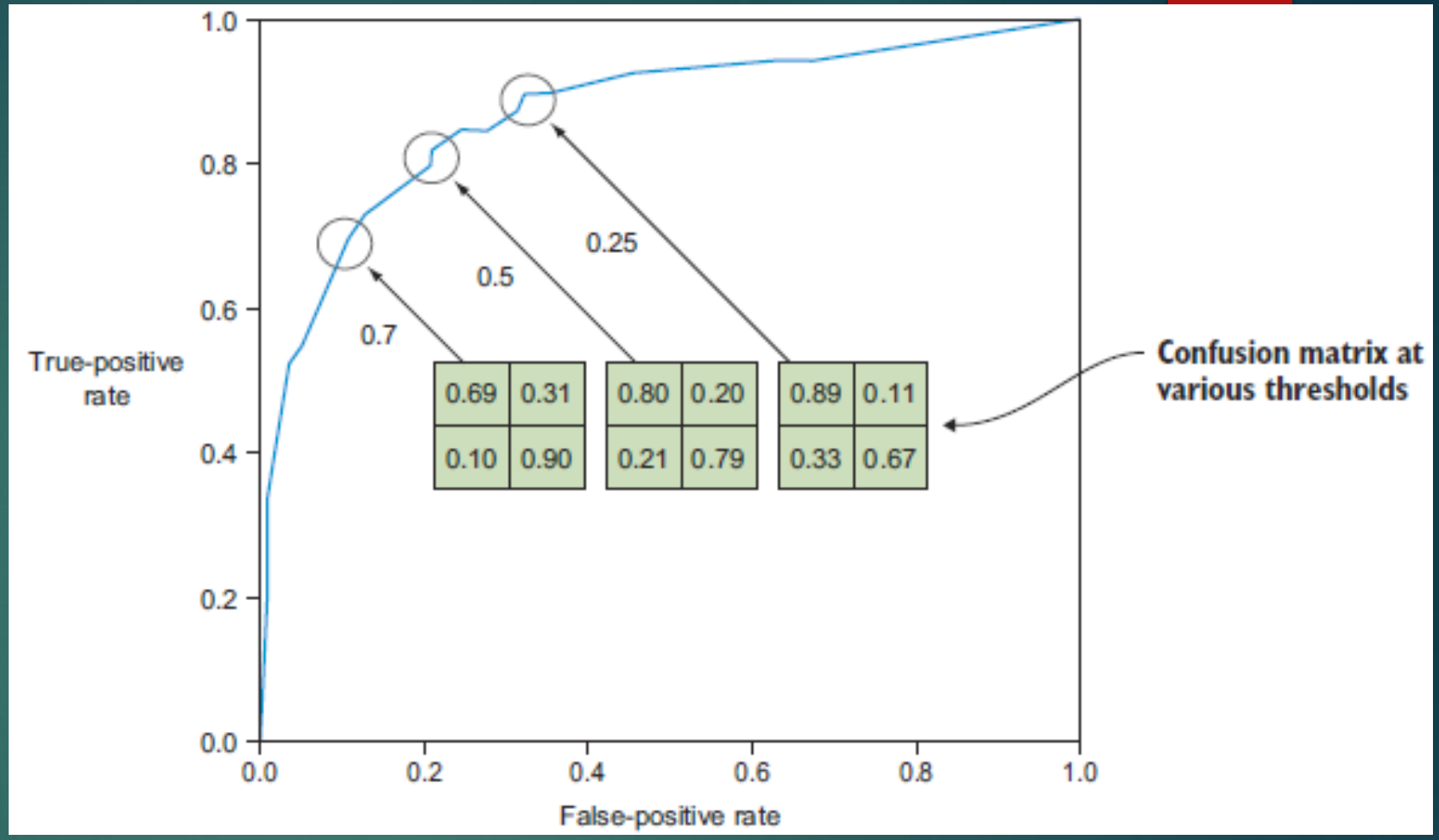
	Survived	Died
308	0.705	0.295
215	0.703	0.297
217	0.700	0.300
54	0.698	0.302
169	0.698	0.302

Threshold: "survived" probabilities > 0.7

A subset of probabilistic predictions from the Titanic test set. After sorting the full table by decreasing survival probability, you can set a threshold and consider all rows above this threshold as survived, otherwise not (died).

Cont.

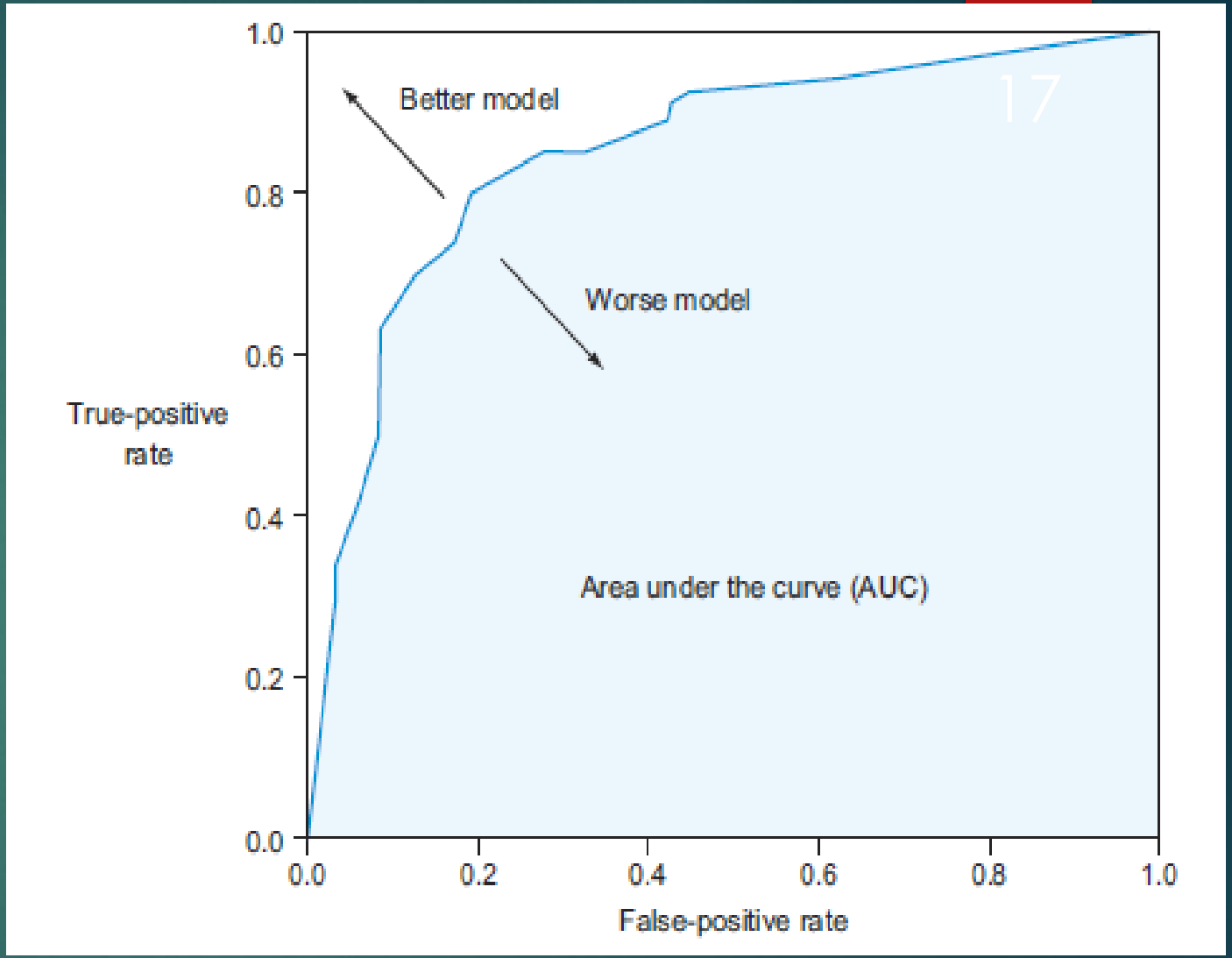
- This trade-off is the "no free lunch" of machine learning because you're able to sacrifice the fraction of instances that you classify correctly for more certainty that you're correct, and vice versa
- patient has cancer or not
- spam filters



The ROC curve defined by calculating the confusion matrix and ROC metrics at 100 threshold points from 0 to 1. By convention, you plot the false-positive rate on the x-axis and the true-positive rate on the y-axis.

Cont.

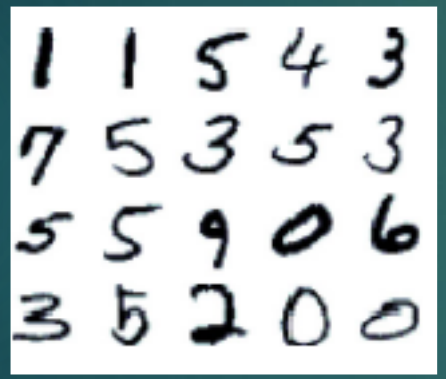
The ROC curve illustrates the overall model performance. You can quantify this by defining the area under the curve or AUC metric: the area under the ROC curve.



Cont.

- 4.2.3 Multiclass classification

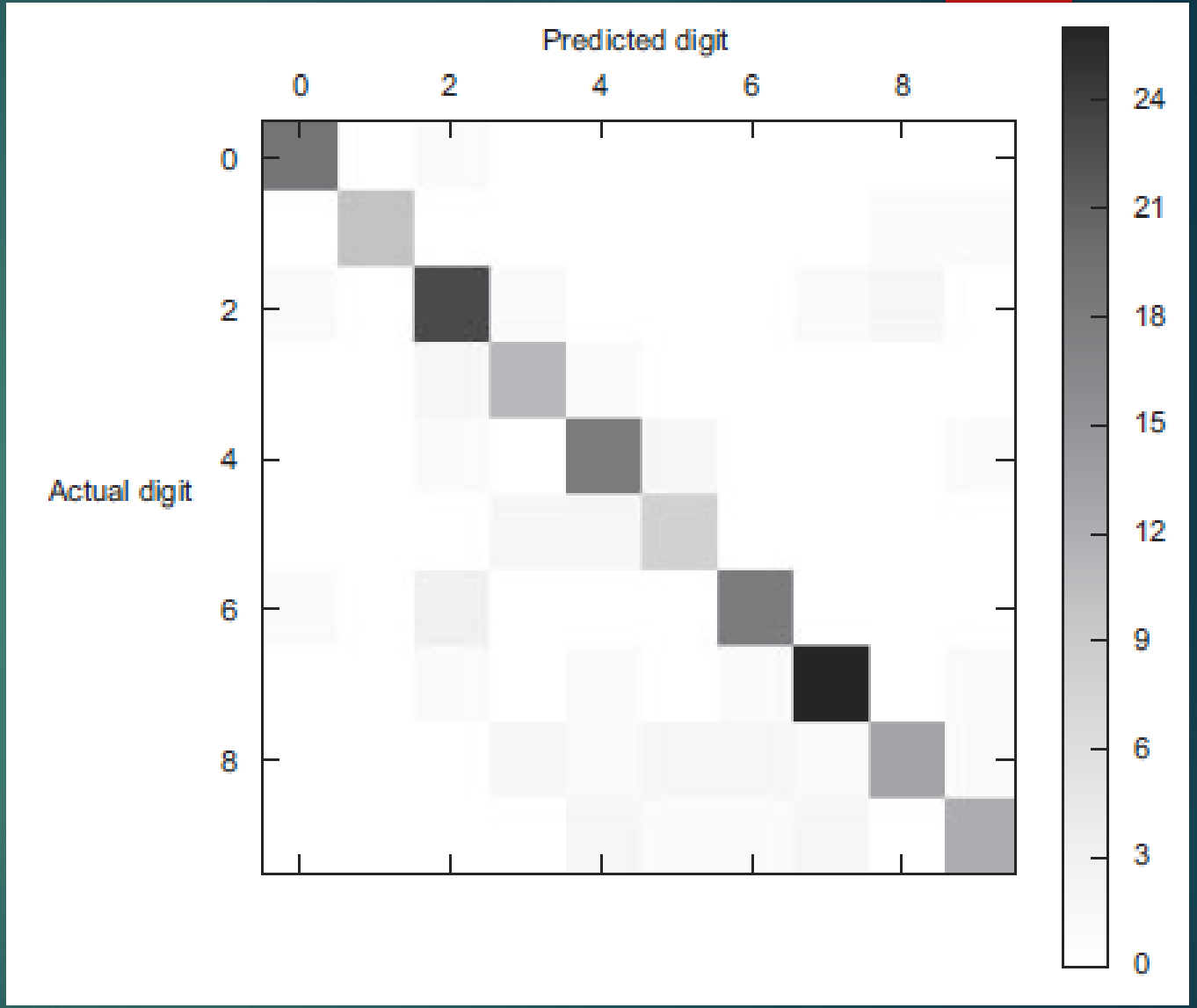
A well-known multiclass classification problem is that of handwritten digit recognition.



Handwritten digits in the MNIST dataset. The entire dataset consists of 80,000 such digits, each in a 28 x 28–pixel image. Without any image processing, each row of our dataset then consists of a known label (0 to 9)and 784 features (one for each of the 28 x 28 pixels).

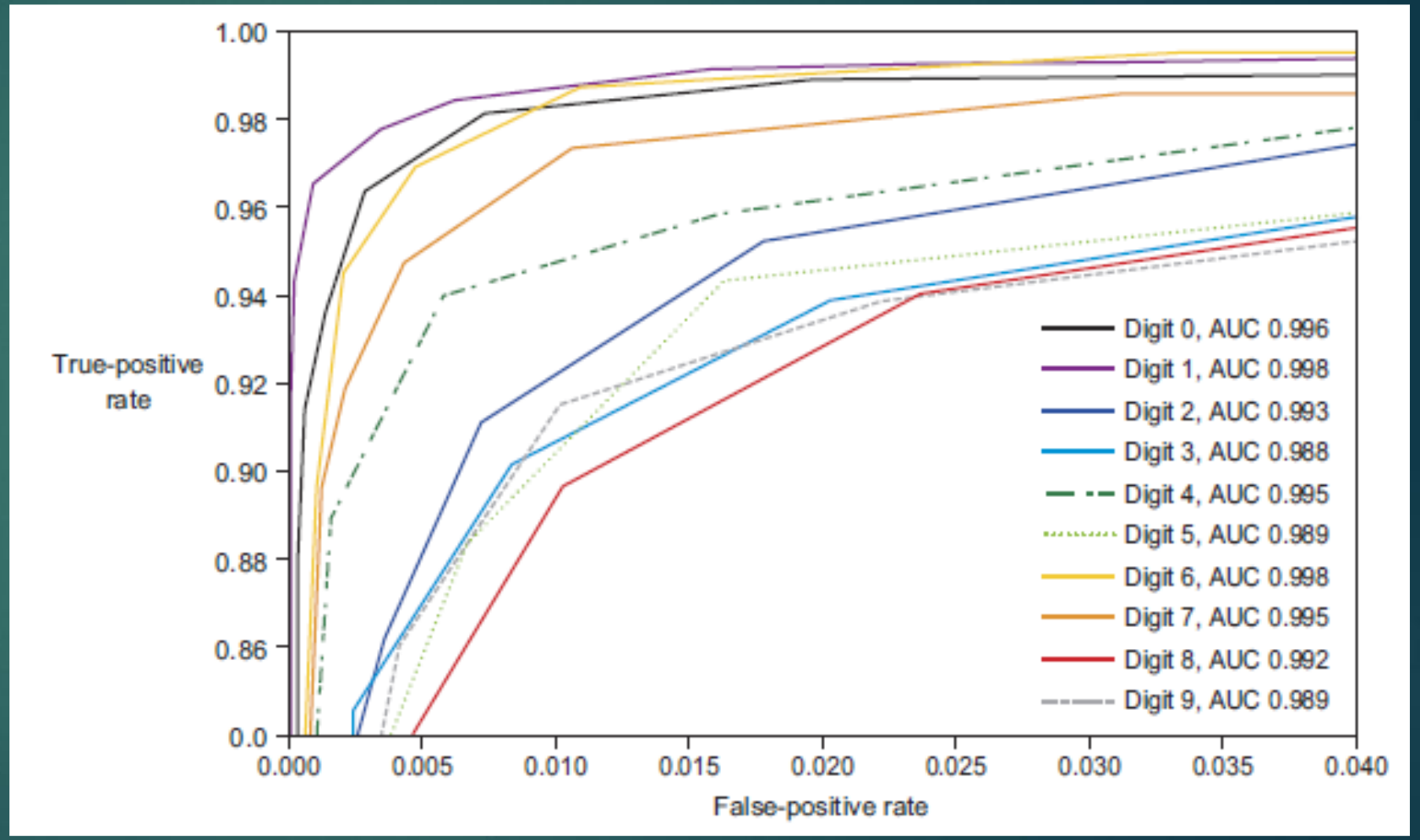
Cont.

- You use the random forest algorithm (introduced in chapter 3) to build a classifier from the training set, and you generate the confusion matrix from the held-out testing set.
- you can easily define it for multiple classes, as every element in the matrix is the class on the row versus the class on the column.



Cont.

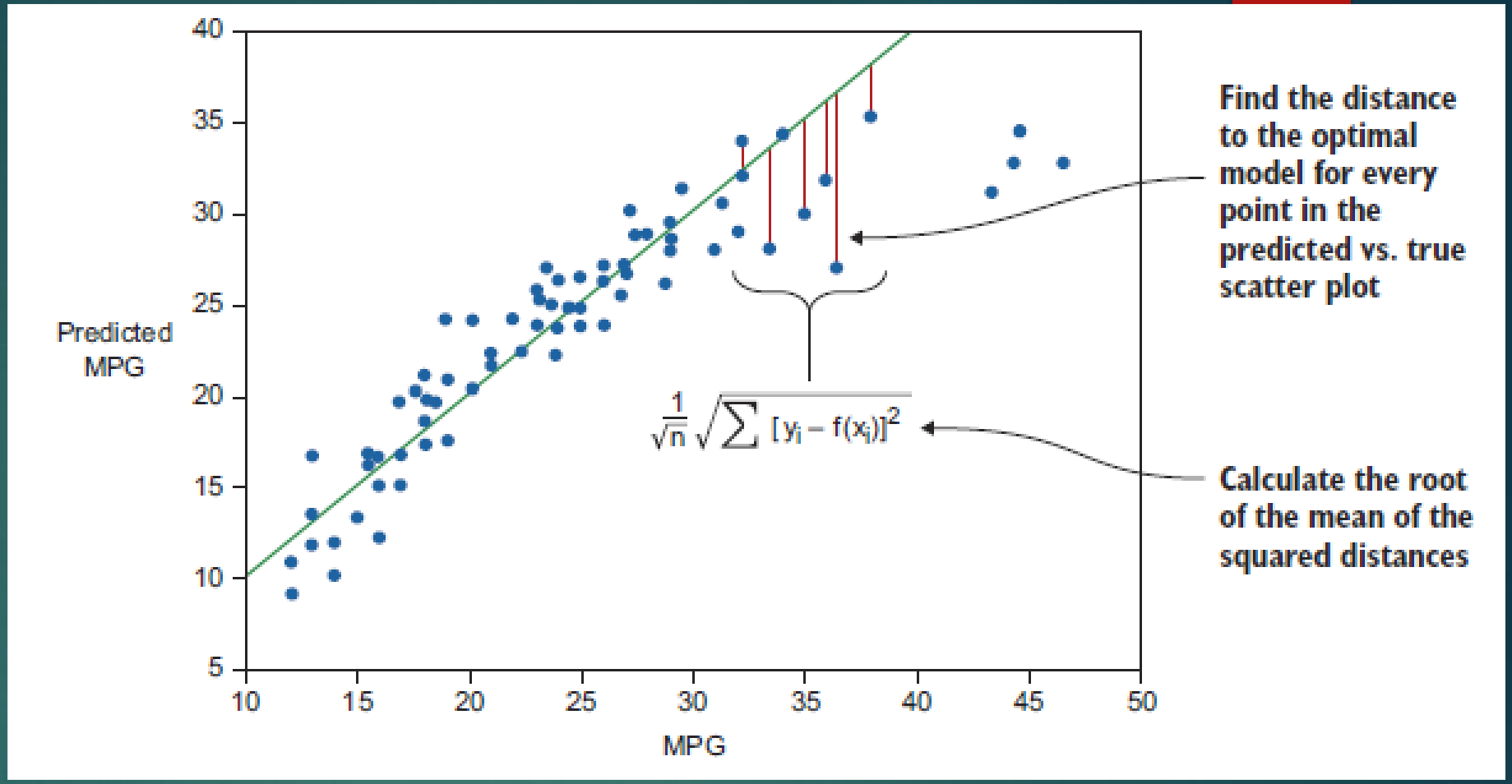
- So how do you generate the ROC curve for multiclass classifiers?
- The ROC curve is in principle applicable to only binary classification problems
- To simulate binary classification in a multiclass problem, you use the one-versus-all trick.
- For each class, you denote the particular class as the positive class, and everything else as the negative class.
- The confusion matrix, however, is generated from the most probable class predictions, whereas the ROC curve shows the performance of the class at all probability thresholds.



Cont.

- 4.3 Evaluation of regression models

4.3.1 Using simple regression performance metrics



The advantage of RMSE is that the result is in the same units as the values themselves, but it's also a disadvantage in the sense that the RMSE value depends on the scale of the problem, and thus isn't easily comparable across datasets.

To overcome this, often it's worthwhile to also compute the R^2 metric, whose response is relative and always in the 0–1 range. 1-(RSS/TSS) .. https://www.geeksforgeeks.org/ml-r-squared-in-regression-analysis/

Cont.

- Whether using MSE, RMSE, or R-squared as the evaluation metric, you should always keep the following in mind:

(A) Always use cross-validation to assess the model.

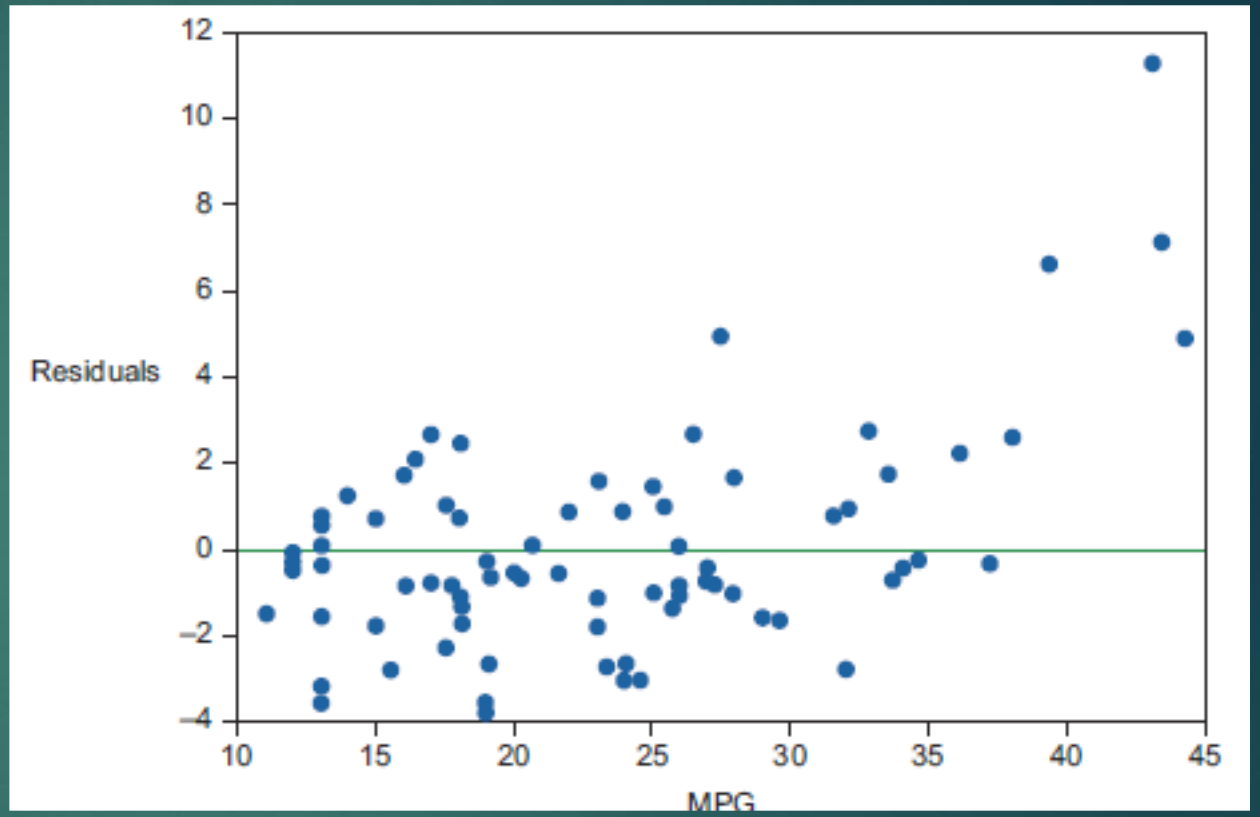
(B) Wherever possible, the evaluation metric should align with the problem at hand.

For instance, if predicting MPG from automobile features, an RMSE of 5 means that you expect the average prediction to differ from the true MPG by 5 miles per gallon.

In addition, regression uses lots of other evaluation metrics, many of which have built-in penalization for overfitting (and thus don't require cross-validation). Examples include the Akaikie information criterion (AIC) and Bayesian information criterion (BIC): look them up!

Cont.

- 4.3.2 **Examining residuals**
- In the previous section, you saw how the residuals, the distance between the predicted and actual values, were used for both of the simple metrics introduced. These residuals can also be interesting to analyze visually themselves.



Cont. (model opt. via parameter tuning)

- 4.4.1 ML algorithms and their tuning parameters

Here are the standard tuning parameters for some of the popular classification algorithms

- Logistic regression—None
- K-nearest neighbors—K, the number of nearest neighbors to average
- Decision trees—Splitting criterion (e.g., gini index, entropy), max depth of tree (related to minimum samples needed to make a split)
- Kernel SVM—Kernel type, kernel coefficient, penalty parameter
- Random forest—Number of trees, number of features to split in each node, splitting criterion, minimum samples needed to make a split
- Boosting—Number of trees, learning rate, max depth of tree, splitting criterion, minimum samples needed to make a split

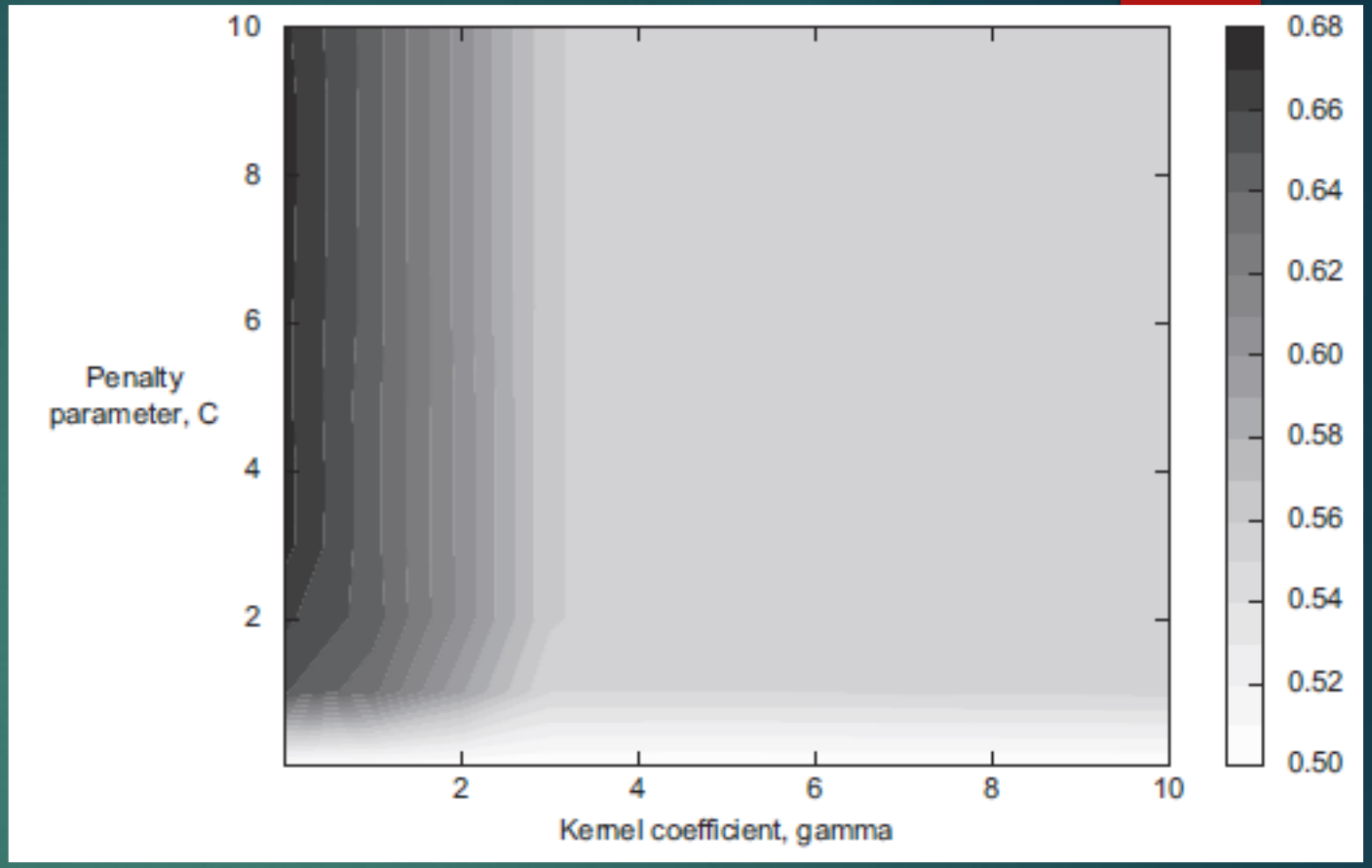
Cont.

- 4.4.2 **Grid search**

1. Choose the evaluation metric that you want to maximize
2. Choose which ML algorithm you want to use
3. Select which tuning parameters you want to optimize over
4. Define the grid as the Cartesian product between the arrays of each tuning parameter.
5. For each combination of tuning parameters in the grid, use the training set to perform cross-validation (using either the hold-out or k-fold-CV method) and compute the evaluation metric on the cross-validated predictions.
6. Finally, select the set of tuning parameters corresponding to the largest (or lowest) value of the evaluation metric. This is the **optimized** model.

Cont.

- You find with the Titanic dataset that the **maximum** cross-validated AUC is 0.670, and it occurs at the tuning parameter vector (gamma = 0.01, C = 6). A few things jump out from the figure:
- The maximum occurs at the **boundary** of the grid
- A high amount of **sensitivity** exists in the accuracy of the predictions to the numerical value of the gamma parameter
- The maximum value occurs **near** gamma = 0, so expressing the grid on a log scale (for example, 10-4, 10-3, 10-2, 10-1) is sensible.
- There's **not** much sensitivity of the AUC as a function of C



Note that the actual best value might have landed somewhere between the values of the grid!

Summary

> When you evaluate models, you **can't double-dip** the training data and use it for evaluation as well as training.

> **Cross-validation** is a more robust method of model evaluation.

> **Holdout cross-validation** is the simplest form of cross-validation, then there k-fold x-valid.

> The basic **model-evaluation workflow** is as follows:

1. Acquire and preprocess the **dataset** for modeling
2. Build models and make **predictions** by using either the holdout or k-fold cross-validation
3. **Evaluate** the predictions with the performance metric of choice
4. Tweak the data and model until the desired model **performance** is obtained

> for **classification** models, we introduced a few model-performance metrics: counting accuracy, the confusion matrix, receiver-operator characteristics (ROC) curve

> for **regression** models, we introduced the root-mean-square error and R-squared estimators. Simple visualizations include: prediction-versus-actual target value scatter plot, and the residuals plot for the set of data points

> You can use a **grid-search** algorithm to optimize a model with respect to tuning parameters.