CO3091 - Computational Intelligence and Software Engineering

Lecture 17

Evaluation Procedures for Machine Learning Approaches

Leandro L. Minku

Image from: http://www.teachhub.com/sites/default/files/styles/large/public/smiley%20face%20options.jpg

Overview

- Evaluation Functions
- Overfitting and Noise
- Choosing Machine Learning Approaches and Parameters
  - Holdout
  - Repeated Holdout
  - Cross-Validation
  - Repeated Cross-Validation
  - Stratification
- Testing a model

Error / Evaluation Functions

- Predictive models can make mistakes (errors).
- We want to minimise these mistakes.
- Mistakes done when predicting a set of data can be measured based on an error / evaluation function.
  - Training.
  - Choosing machine learning approach or parameters.
- From the problem point of view, the goal of machine learning is to create models able to generalise to unseen data.
  - This cannot be calculated at training time.

We need to estimate the error based on a known data set.

Image from: http://vignette2.wikia.nocookie.net/fantendo/images/2/25/Mario_Artwork_-_Super_Mario_3D_World.png/revision/latest?cb=20131025223058

Examples of Evaluation Functions Using a Known Data Set

- Classification error:
  - Given a data set D with examples $(\mathbf{x}_i, y_i)$, $1 \leq i \leq m$.
  - The actual output (target) for $\mathbf{x}_i$ is y_i.
  - The prediction given by a classification model to $\mathbf{x}_i$ is y_i'.
  - y_i and y_i' are categorical values.

$$\text{Classification error} = \frac{1}{m}\sum_{i=1}^{m}(y_i \neq y_i')$$

Number of misclassified examples

- Classification accuracy:

Classification accuracy = 1 - classification error

Examples of Evaluation Functions Using a Known Data Set

- Mean Absolute Error (MAE):
  - Given a data set D with examples $(\mathbf{x}_i, y_i)$, $1 \le i \le m$.
  - The actual output (target) for $\mathbf{x}_i$ is y_i.
  - The prediction given by a regression model to $\mathbf{x}_i$ is y_i'.
  - y_i and y_i' are numerical values.

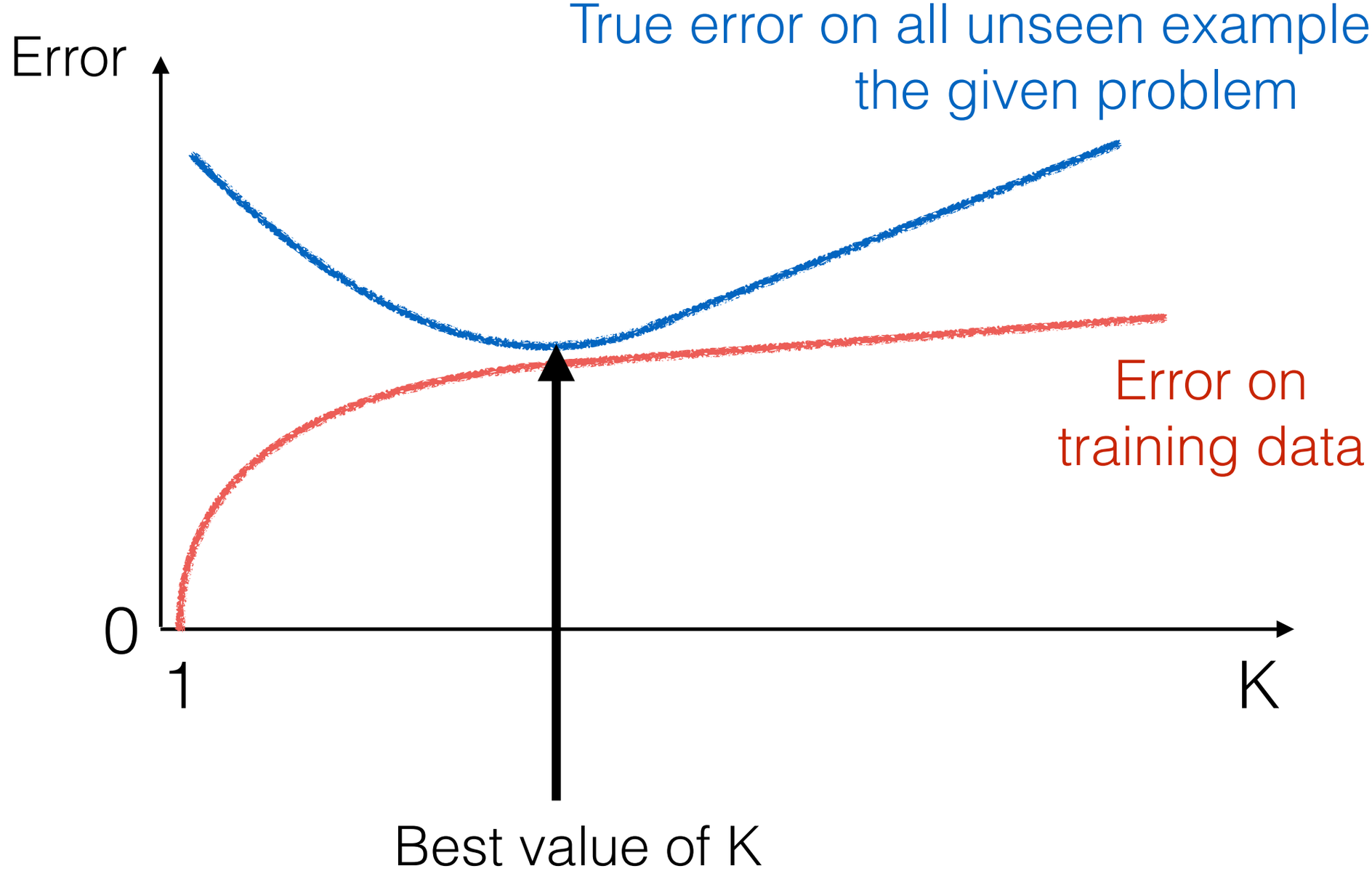
$$\text{MAE} = \frac{1}{m}\sum_{i=1}^{m} |y_i - y_i'|$$

- Mean Squared Error (MSE): $\text{MSE} = \frac{1}{m}\sum_{i=1}^{m} (y_i - y_i')^2$
- Root Mean Squared Error (RMSE): $\text{RMSE} = \sqrt{\text{MSE}}$

What Data Set to Use for Evaluating an Approach or Parameter?

- The data used for training / building a model is referred to as the training set (training error).
- However, if we concentrate only on minimising the training error, we may get poor results on unseen data.

Typical Impact of k-NN's Parameter k



Example with WEKA

```
java -cp myweka.jar:weka.jar:junit-4.12.jar  weka.gui.GUIChooser
```

```
java -cp myweka.jar:weka.jar weka.classifiers.lazy.MyKnnSolution -K 1
-t breast-cancer-wisconsin-nomissing.arff -split-percentage 66 -s 1
```

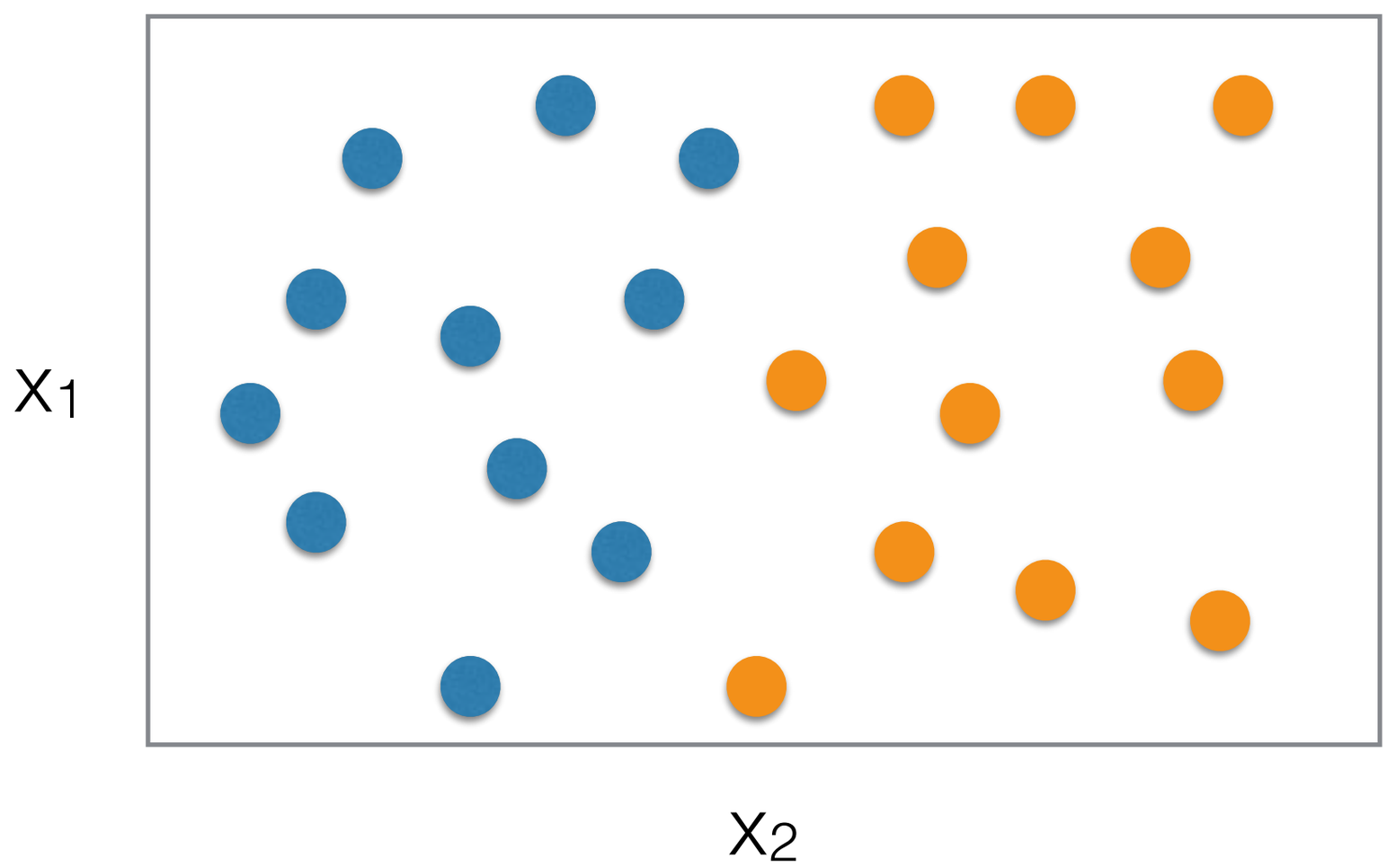
```
java -cp myweka.jar:weka.jar weka.classifiers.lazy.MyKnnSolution -K 3
-t breast-cancer-wisconsin-nomissing.arff -split-percentage 66 -s 1
```

```
java -cp myweka.jar:weka.jar weka.classifiers.lazy.MyKnnSolution -K 10
-t breast-cancer-wisconsin-nomissing.arff -split-percentage 66 -s 1
```

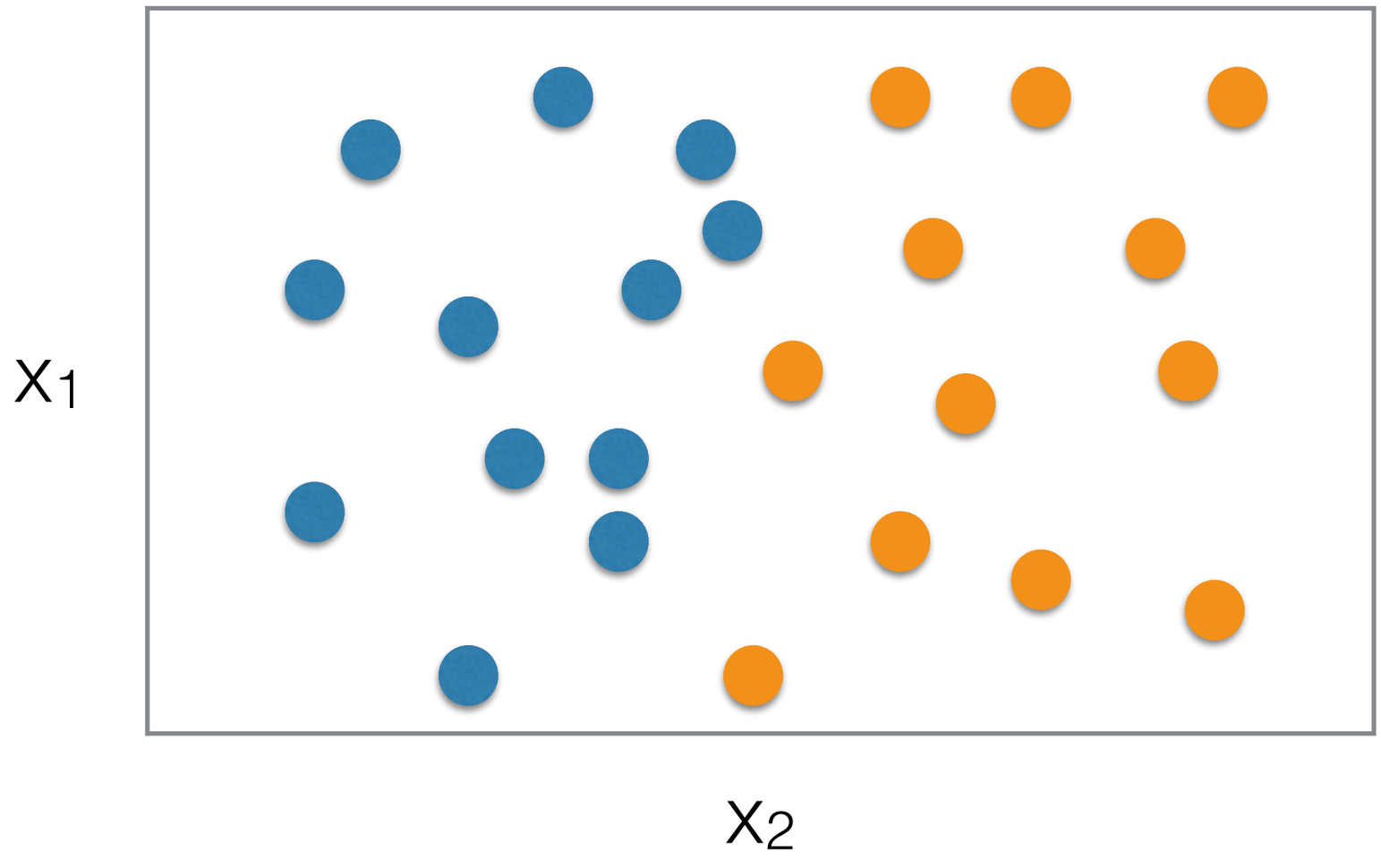
What Data Set to Use for Evaluating an Approach or Parameter?

- Why concentrating only on the training error may result in poor results on unseen data?
  - Real world data sets frequently have some noise, i.e., measurement error when collecting the data.

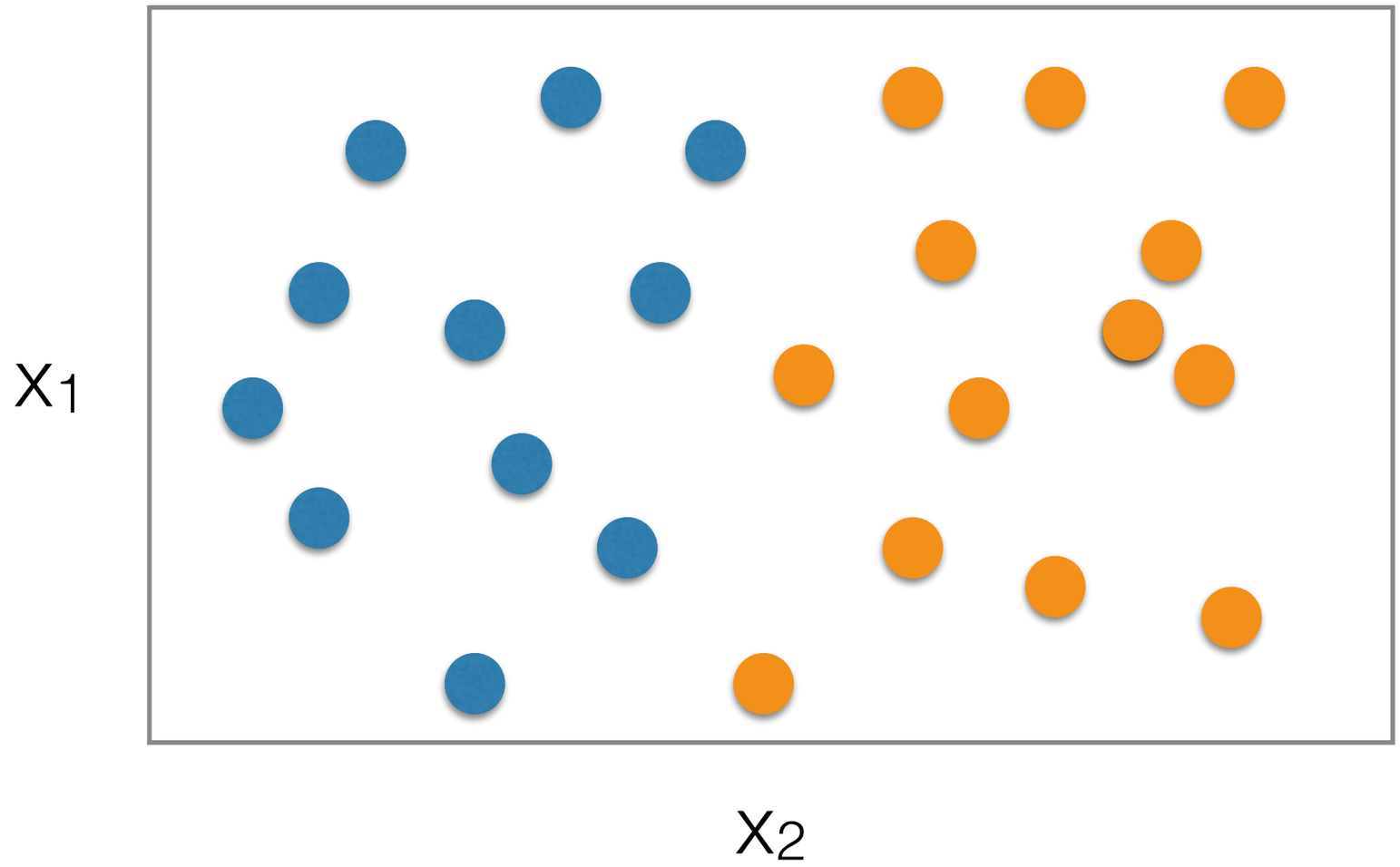
Noise



Noise



Noise



What Data Set to Use for Evaluating an Approach or Parameter?

- Why concentrating only on the training error may result in poor results on unseen data?
  - Real world data sets frequently have some noise, i.e., measurement error when collecting the data.
  - If we only concentrate on minimising the error on the training data, we are likely to learn noise, i.e., wrong information.

Typical Impact of k-NN's Parameter k

Class boundaries

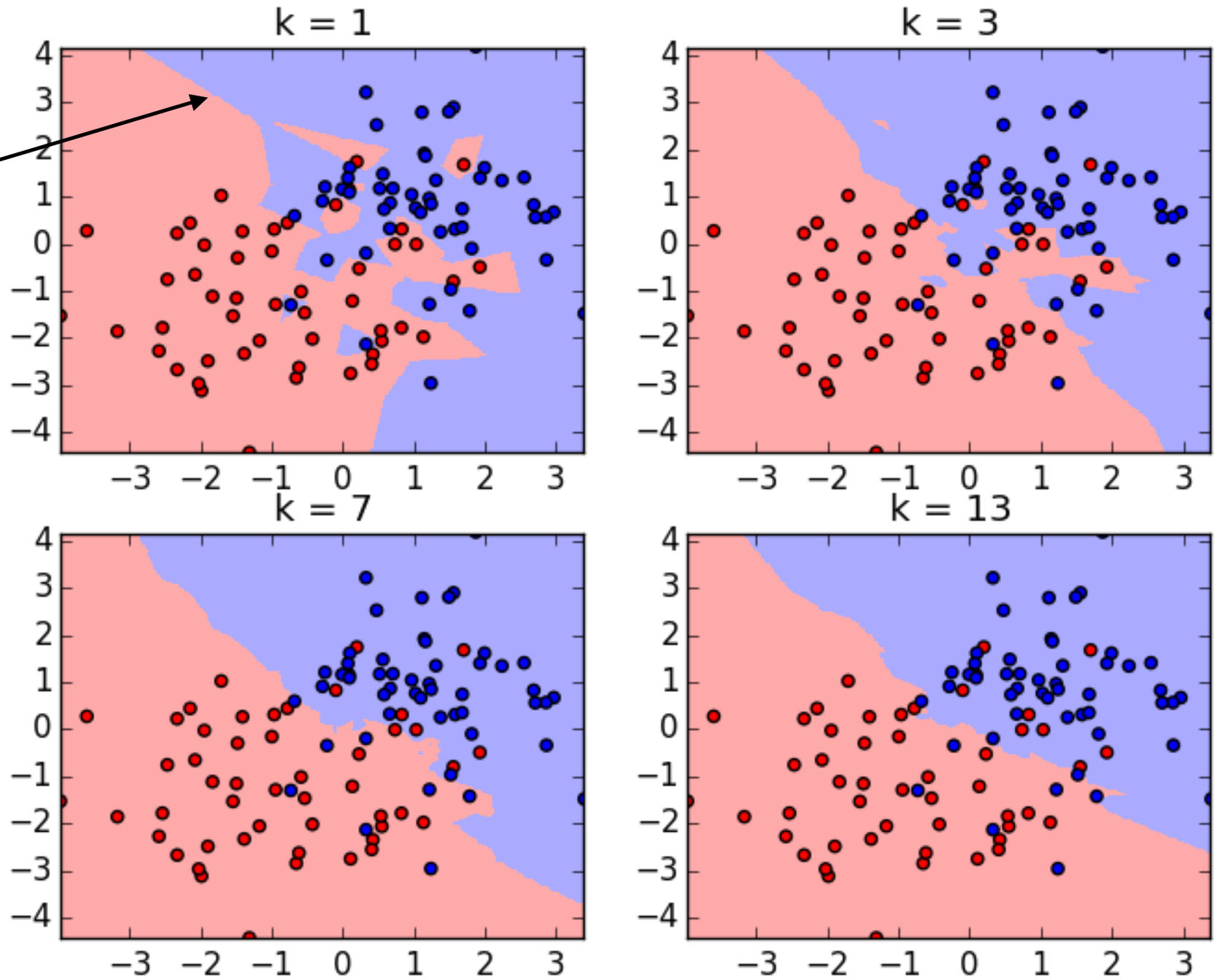


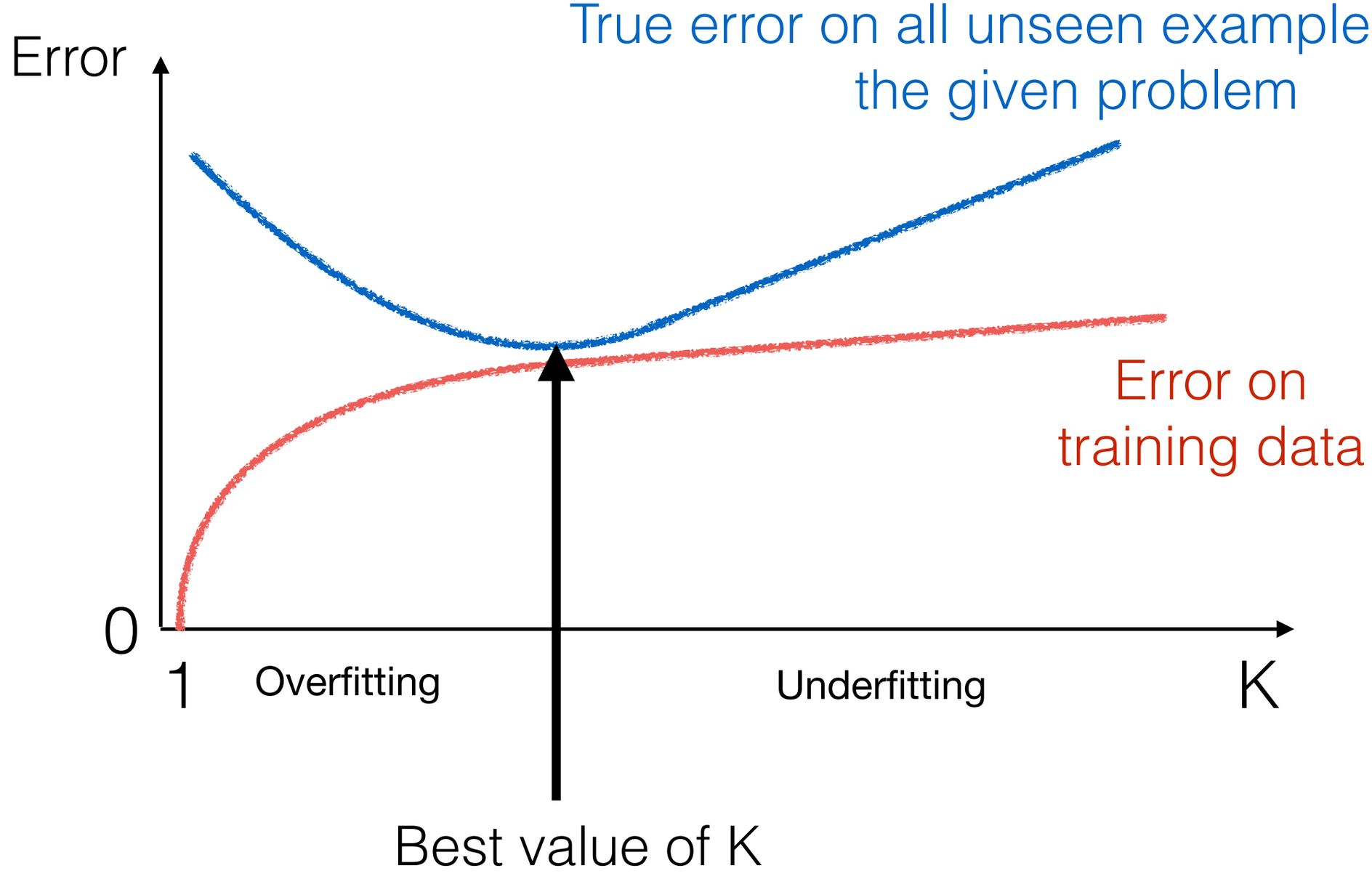
Image from: http://ljdursi.github.io/ML-for-scientists/outputs/classification/knn-vary-k.png

Overfitting

If we only concentrate on minimising the error on the training data, we may be learning noise, i.e., wrong information.

Training error will be very low, but error on unseen data will be high.

Typical Impact of k-NN's Parameter k



What Data Set to Use for Evaluating an Approach or Parameter?

- We could split the available labelled data into two separate sets:
  - Training set: used by the machine learning approach to learn a model.
    - Machine learning approach may not only try to minimise the error on the training set, but also adopt some procedure to improve generalisation.
  - Validation set: used to choose between different machine learning approaches (or parameters for the approaches).
    - It estimates the error on data unseen at the time of building the model.

Training and Validation

Training Set

x1 = age	x2 = salary	x3 = gender	…	y = good/ bad payer
18	1000	female	…	Good
30	900	male	…	Bad
20	5000	female	…	Good
…	…	…	…	…

→ Machine Learning Algorithm →

Predictive Model

Validation Set

x1 = age	x2 = salary	x3 = gender	…	y = good/ bad payer
18	1100	male	…	Good
30	1500	male	…	Bad
20	5000	male	…	Good
…	…	…	…	…

→

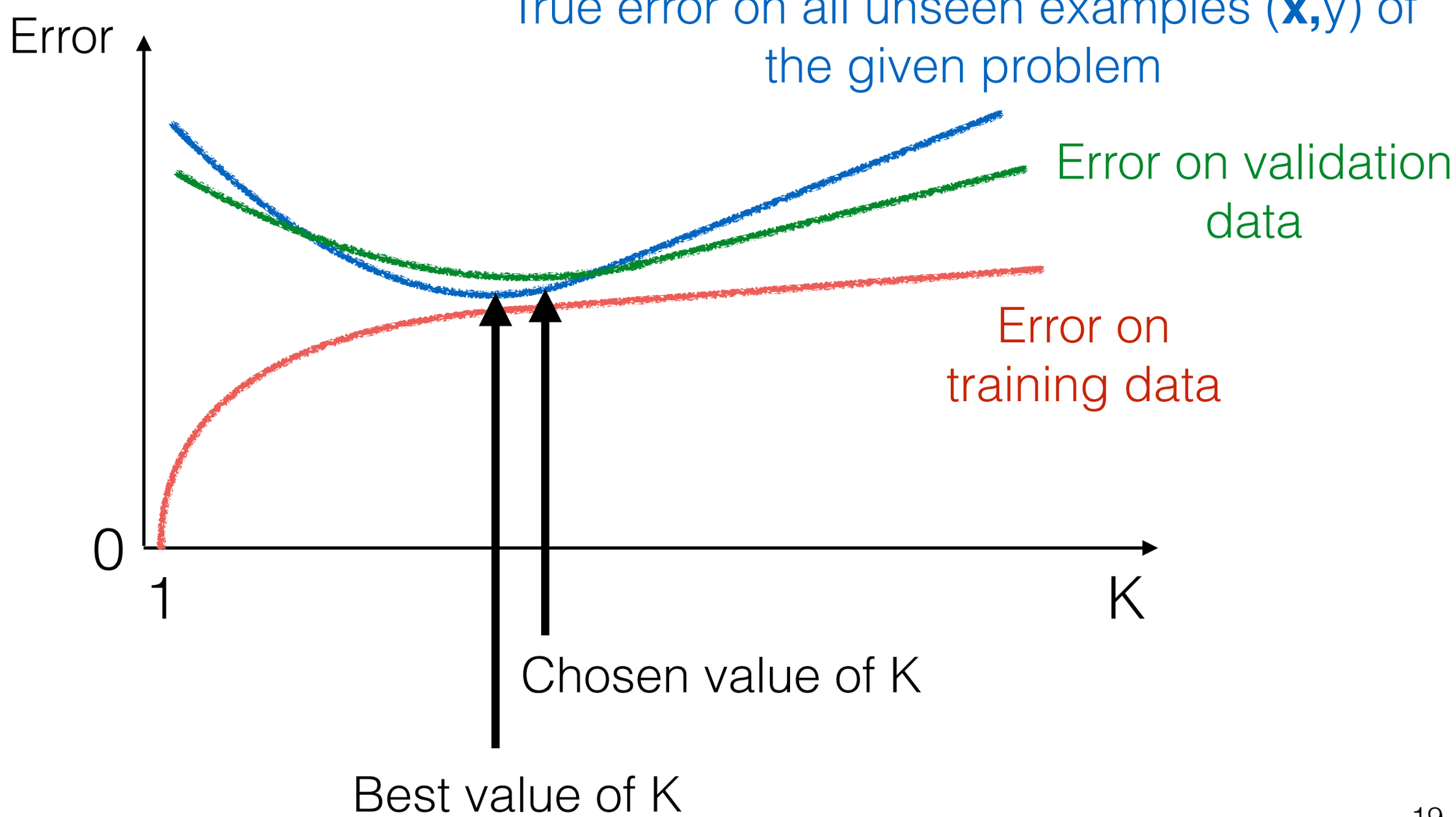
→ Predictions

↓

Validation error

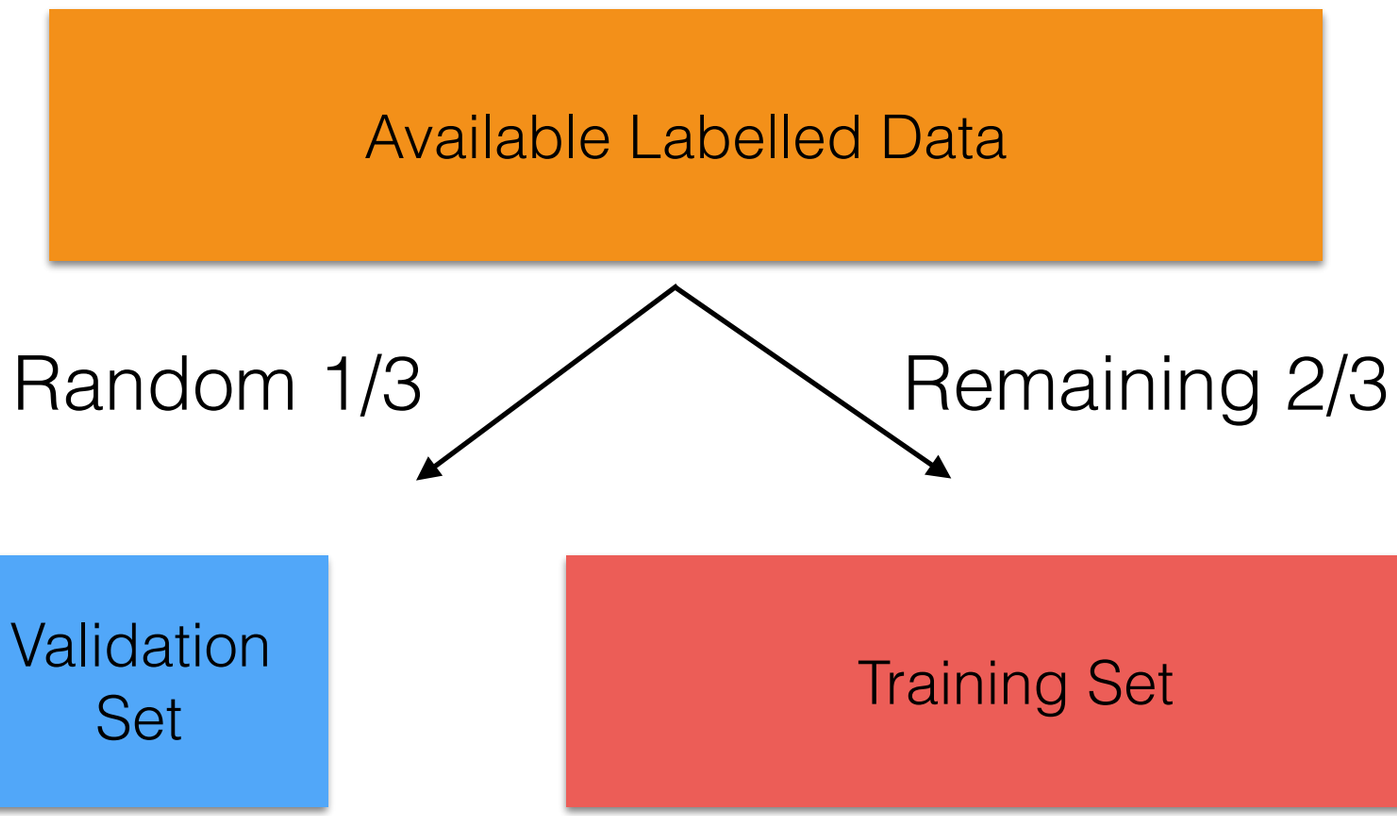
You can choose to use the machine learning approach or parameters that lead to the lowest validation error.

Typical Impact of k-NN's Parameter k



How to create the splits between training and validation data?

Holdout



In WEKA, -split-percentage 66 -s 1 will use 66% of data for training and 34% for validation using random seed 1.

- PS: WEKA terminology does not distinguish between validation and test set.

Repeated Holdout

- Problem of holdout: different training and validation sets will lead to different results.
  - A given approach / parameter may be lucky on a certain partition.
- Repeated holdout:
  - In order to choose a machine learning approach or parameter, repeat the holdout process several times (with different random seeds) to create different training / validation partitions.

Repeated Holdout

- Repeat a given number of times r (e.g., r=30):
  - [Choose a random seed that has not been used in any previous iteration]
  - Pick 1/3 of the data uniformly at random to compose the validation set.
  - Use the remaining 2/3 for training.
  - Calculate the error using the validation data.

- Use the average or median of the r validation errors as a measure for choosing a machine learning approach / parameter.

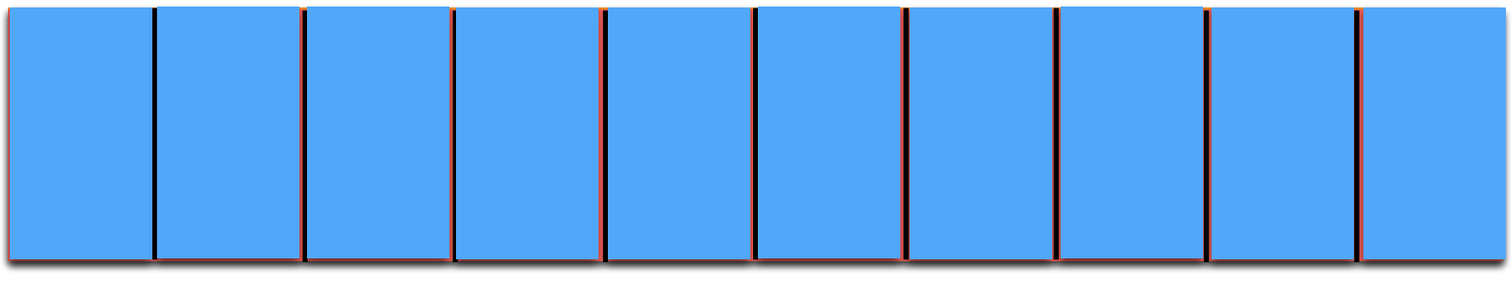
Repeated Holdout

- Problem of repeated holdout:
  - Not all the available examples will have been used for training in at least one run.
  - Not all the examples will have been used for validation in at least one run.

K-Fold Cross-Validation

- Divide the available data into K folds (e.g., K=10).
- For each fold, use it for validation and the remaining for training.
- Use the average or median of the K validation errors as the measure for choosing a machine learning approach or parameter.

Validation data Training data



Val error 1 | Val error 2 | Val error 3 | Val error 4 | Val error 5 | Val error 6 | Val error 7 | Val error 8 | Val error 9 | Val error 10

Stratification in Classification Problems

- It may happen that certain partitions do not represent well all classes.
  - E.g., a certain training partition may not contain any example of a given class. So, it would not be able to learn this class.
- Holdout and K-fold cross validation can be combined with stratification.
  - E.g., for the cancer data, if your training set contains 66% of the examples, pick 66% of the benign examples and 66% of the malignant examples to compose it.
- Stratification can help us to get all classes represented in all training and validation sets.

[Example of K-fold cross-validation using WEKA explorer]

Repeated K-Fold Cross-Validation

- Problem of K-Fold Cross-Validation:
  - Different orders of the available data will still lead to different partitions of training and validation sets.

- Repeated K-Fold Cross-Validation:
  - Repeat K-Fold Cross-Validation r times, e.g., r=10.
  - Use the average or median of the r * K validation errors as the evaluation measure.

[Example using WEKA experimenter]

What Data Set to Use for Estimating the Error of a Model on Unseen Data?

- Training error is used to train (build) a model using a machine learning approach and parameter values.
- Validation error is used for choosing a machine learning approach or parameters.
- Once the machine learning approach / parameter has been chosen, we cannot use the training or validation error to provide an estimate of its its resulting model's performance on future unseen data anymore.
  - Using the training error would lead to the problems discussed earlier.
  - The validation error would also be optimistic, because the approach / parameter has been chosen to do well on the validation data.
- Once the machine learning approach / parameter has been chosen, we need to use a data set that has neither been used for training nor for validation in order to estimate its resulting model's generalisation ability.

What Data Set to Use for Estimating the Error of a Model on Unseen Data?

- Test set: separate data set used neither for training nor for validation. It can be used to give an idea of how well the model will perform / is performing in practice, i.e., how good the generalisation to future unseen data is likely to be.

- It may be problematic to create such test set if we have small data.

- Test error is still just an estimate of the true error on all the existing unseen data.

- We can't use the test set to choose between machine learning approaches / parameters, because it will then work as a validation set and will give an optimistic estimate of the generalisation ability.

Given a machine learning prediction problem, how to choose a supervised learning approach and parameters to use?

Possible way: use repeated k-fold cross-validation to choose a machine learning approach and parameters.

Image from: http://www.clipartbest.com/cliparts/dc7/5dR/dc75dRdc9.jpeg

Given a machine learning prediction problem,
how to choose a supervised learning approach and
parameters to use?

Some problems are well understood in the literature,
and you may use the machine learning approach
recommended to them in the literature.

Image from: http://www.clipartbest.com/cliparts/dc7/5dR/dc75dRdc9.jpeg

Given a machine learning prediction problem, how to choose a supervised learning approach and parameters to use?

Still, results may vary depending on your own data. So, you may still wish to validate different approaches / parameters.

Image from: http://www.clipartbest.com/cliparts/dc7/5dR/dc75dRdc9.jpeg

Once you chose an approach, how to get an idea of how well its resulting model will perform in practice, i.e., on future unseen data?

Check its error on a test set which hasn't been used for training or choosing the machine learning approach / parameters.

Image from: http://www.clipartbest.com/cliparts/dc7/5dR/dc75dRdc9.jpeg

Once you chose an approach, how to get an idea of how well its resulting model will perform in practice, i.e., on future unseen data?

However, be careful, because if this test set does not represent well your data space, you may get a bad estimate.

Image from: http://www.clipartbest.com/cliparts/dc7/5dR/dc75dRdc9.jpeg

Further Reading

Ji-Hyun Kim

Estimating classification error rate: Repeated cross-validation, repeated hold-out and bootstrap

Computational Statistics & Data Analysis

Volume 53, Issue 11, 1 September 2009, Pages 3735–3745

http://www.sciencedirect.com/science/article/pii/S0167947309001601

D.J. Hand, H. Mannila, P. Smyth

Principles of Data Mining

MIT Press, 2003

Sections 7.4.4 and 7.5

ftp://gamma.sbin.org/pub/doc/books/Principles_of_Data_Mining.pdf

Lab session at 3pm!