Machine Learning : Process

Dr Alexander A S Gunawan
aagung@binus.edu
http://sigmetris.com
08175001010

IlmuKomputer.Com

BrainMatics

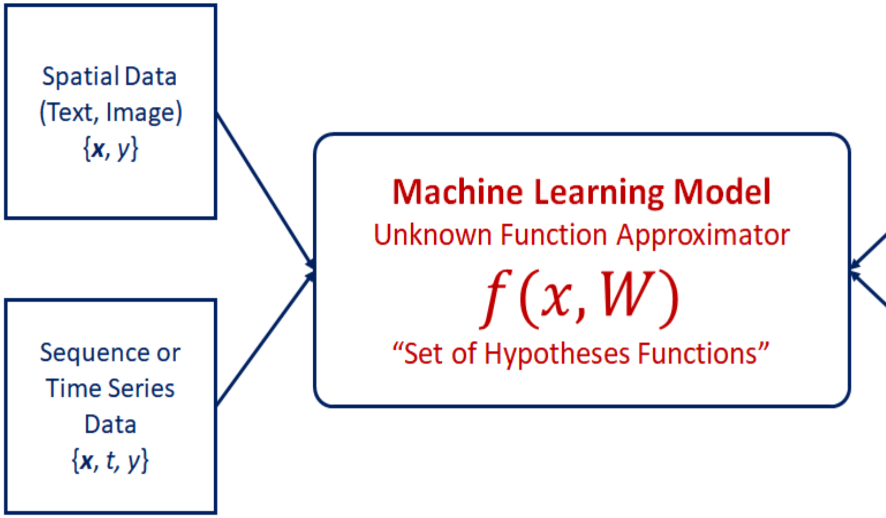
What is machine learning?

- The art *and* science of:
  - Giving computers the ability to learn to make decisions from data
  - ... without being explicitly programmed!
- Examples:
  - Learning to predict whether an email is spam or not
  - Clustering wikipedia entries into different categories
- Supervised learning: Uses labeled data
- Unsupervised learning: Uses unlabeled data

A Typical Machine Learning Process

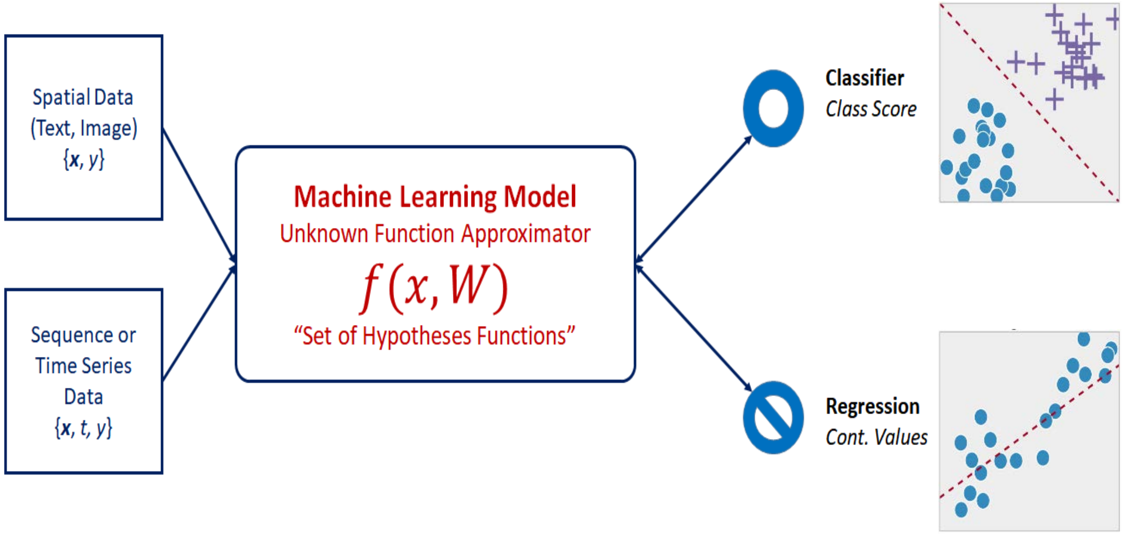
Supervised Learning Tasks

- **Tasks:** Given data (patterns inside without analytic solution), perform classification or regression



Supervised Learning Tasks

- **Tasks:** Given data (patterns inside without analytic solution), perform classification or regression



Classification Problem

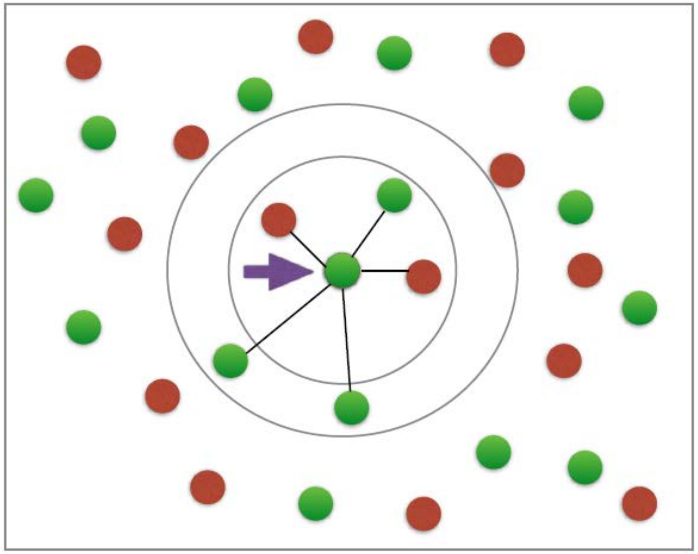
Goal: predict category of new observation

Earlier Observations $\xrightarrow{\text{Estimate}}$ CLASSIFIER

Unseen Data $\xrightarrow{\text{CLASSIFIER}}$ Class

k-Nearest Neighbors

- Basic idea: Predict the label of a data point by
  - Looking at the ‘k’ closest labeled data points
  - Taking a majority vote



Measuring model performance

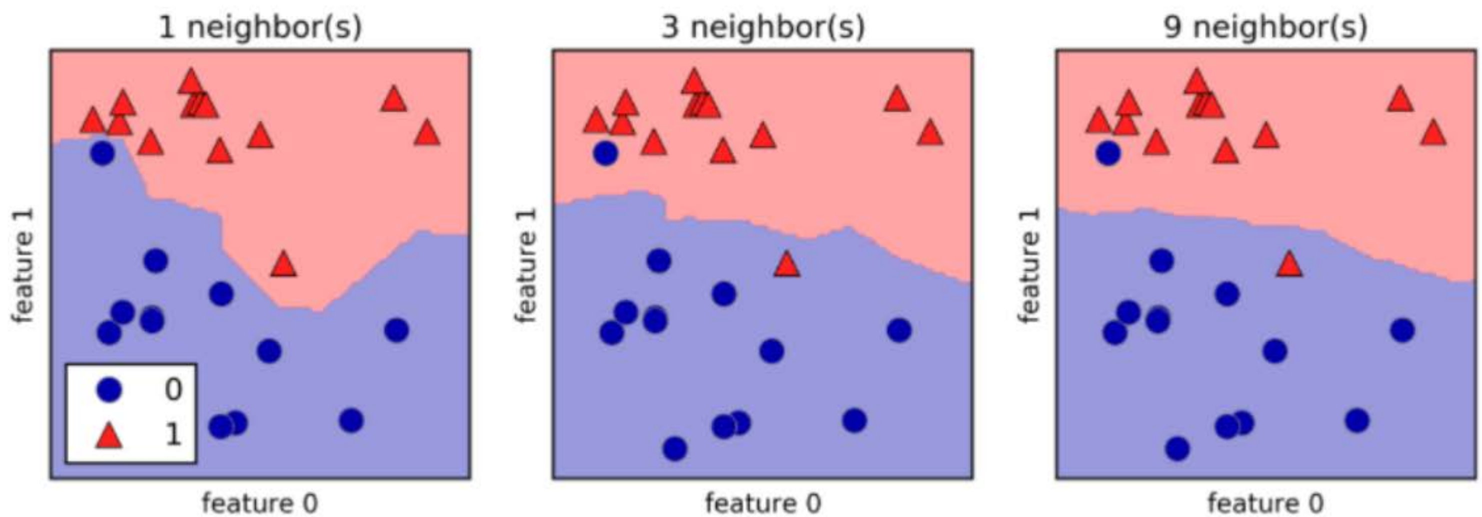
- In classification, accuracy is a commonly used metric
- Accuracy = Fraction of correct predictions
- Which data should be used to compute accuracy?
- How well will the model perform on new data?

Measuring model performance

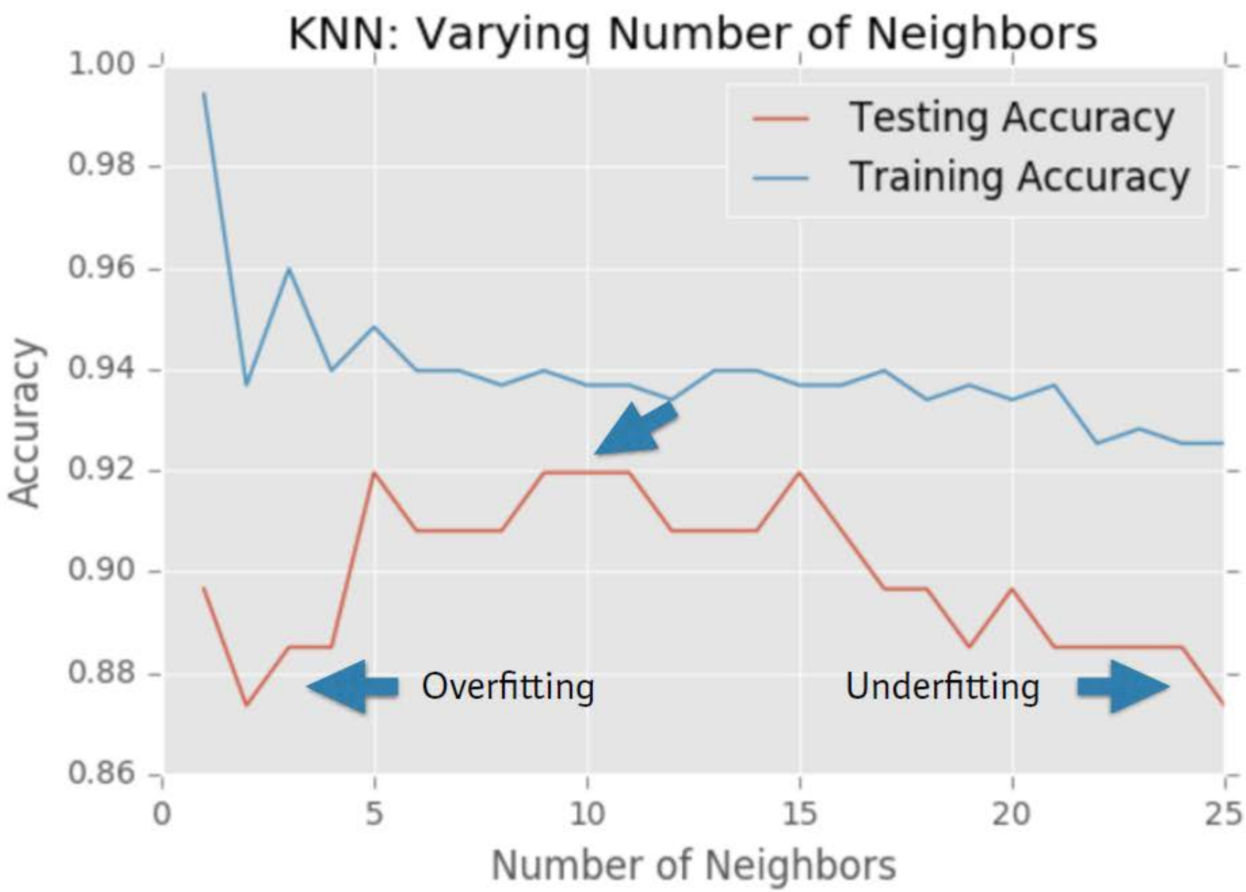
- Could compute accuracy on data used to fit classifier
    - NOT indicative of ability to generalize
- Split data into training and test set
    - Fit/train the classifier on the training set
    - Make predictions on test set
    - Compare predictions with the known labels

Model complexity

- Larger k = smoother decision boundary = less complex model
- Smaller k = more complex model = can lead to overfitting



Model complexity and over/underfitting



Overfitting

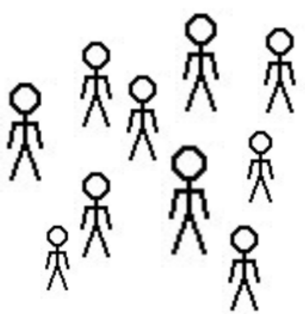
- **Accuracy** will depend on dataset **split** (train/test)
- High **variance** will **heavily** depend on **split**
- **Overfitting** = model fits **training set** a lot better than **test set**
- Too **specific**

Underfitting

- Restricting your model **too much**
- High **bias**
- Too **general**

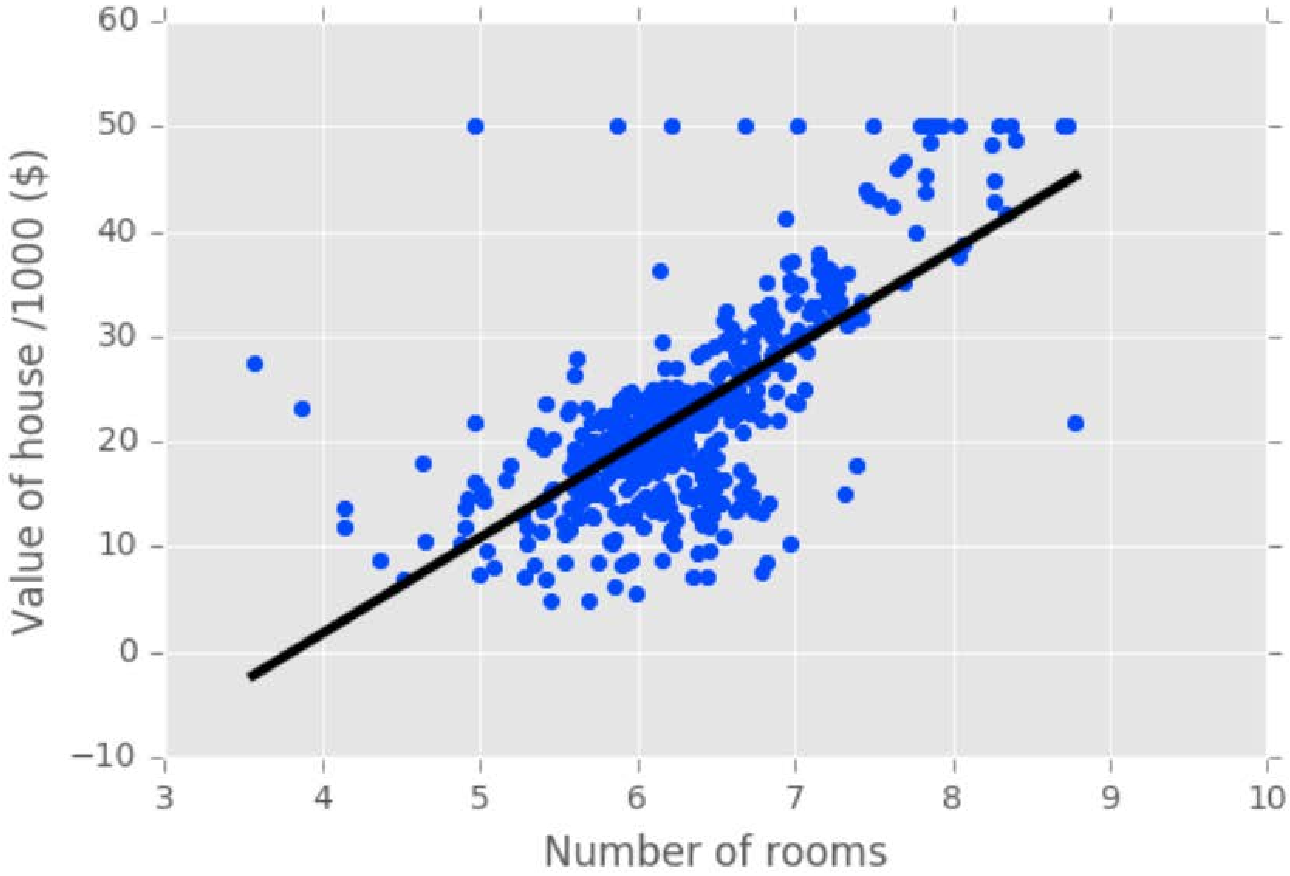
Regression

PREDICTORS → REGRESSION FUNCTION → RESPONSE



- Relationship: **Height - Weight**?
- Linear?
- Predict: Weight → **Height**

Fitting a regression model



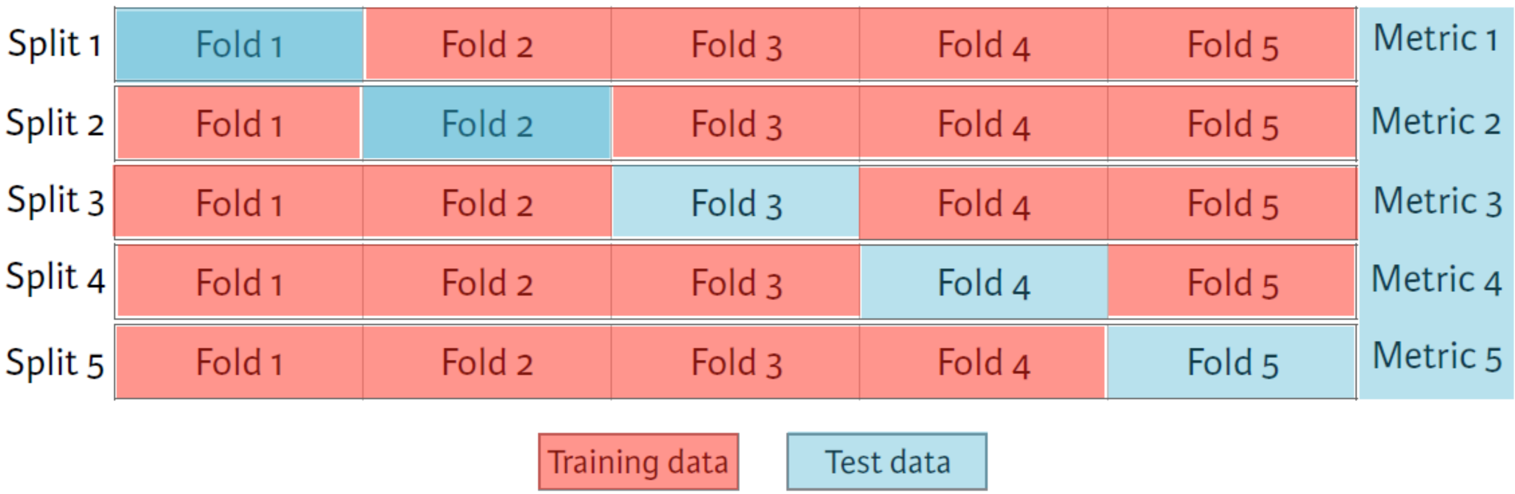
Regression mechanics

- $y = ax + b$
  - y = target
  - x = single feature
  - a, b = parameters of model
- How do we choose a and b?
- Define an error function for any given line
  - Choose the line that minimizes the error function

Cross-validation motivation

- Model performance is dependent on way the data is split
- Not representative of the model's ability to generalize
- Solution: Cross-validation!

Cross-validation basics



Cross-validation and model performance

- 5 folds = 5-fold CV
- 10 folds = 10-fold CV
- k folds = k-fold CV
- More folds = More computationally expensive

Diagnosing classification predictions

- Confusion matrix

	Predicted: Spam Email	Predicted: Real Email
Actual: Spam Email	True Positive	False Negative
Actual: Real Email	False Positive	True Negative

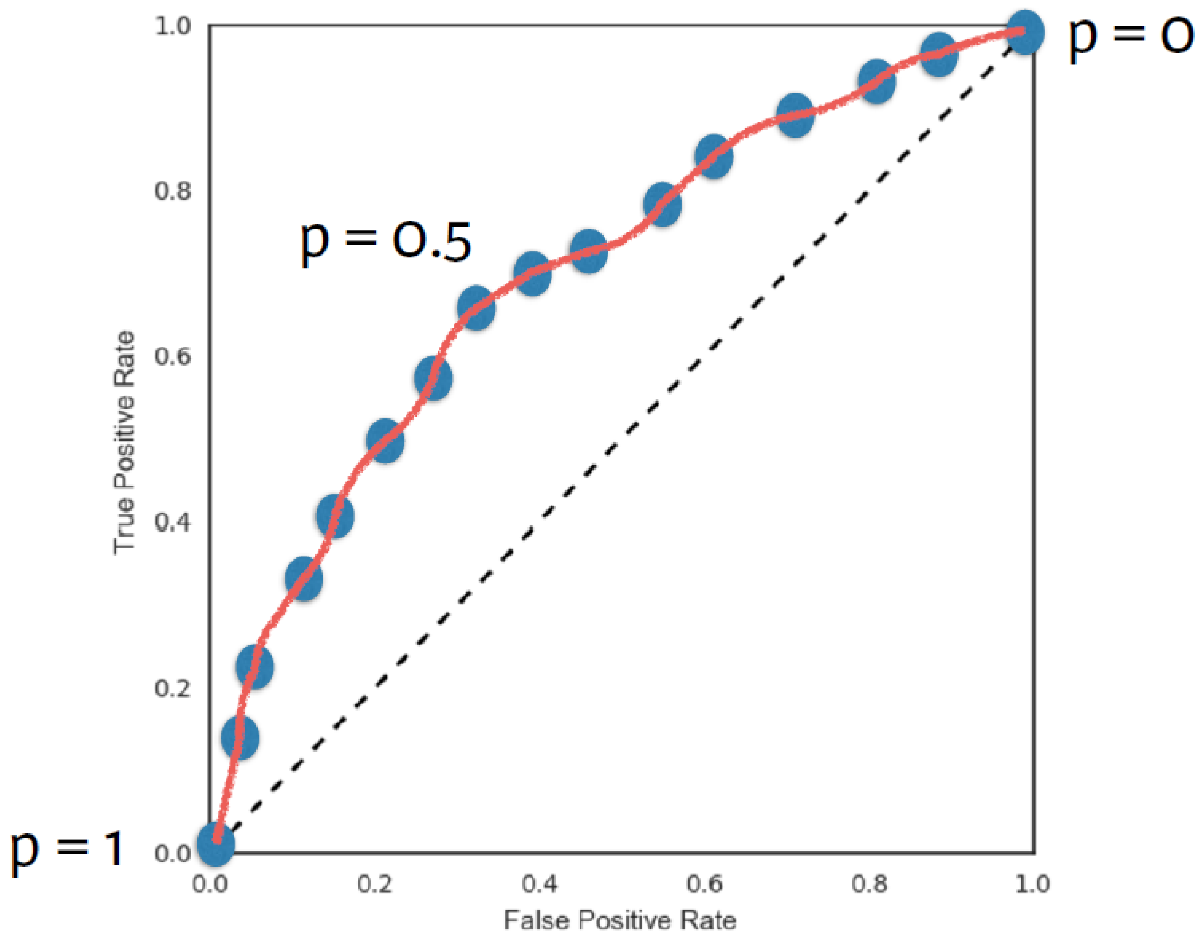
- Accuracy: $$\frac{tp + tn}{tp + tn + fp + fn}$$

Metrics from the confusion matrix

- Precision : $\dfrac{tp}{tp + fp}$
- Recall : $\dfrac{tp}{tp + fn}$
- F1 score : $2 \cdot \dfrac{\text{precision} \cdot \text{recall}}{\text{precision} + \text{recall}}$

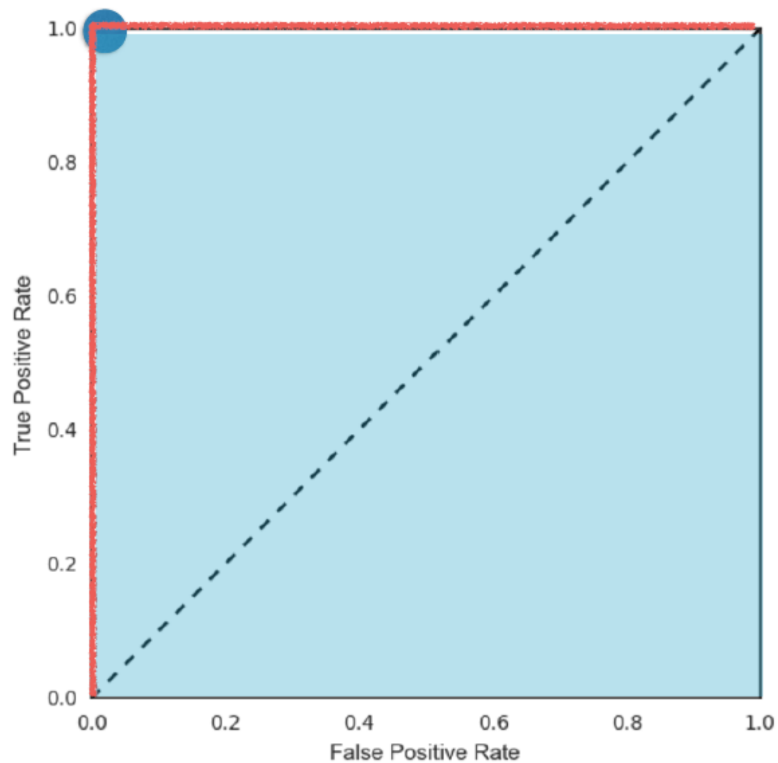
- High precision: Not many real emails predicted as spam
- High recall: Predicted most spam emails correctly

The ROC curve



Area under the ROC curve (AUC)

- Larger area under the ROC curve = better model



Choosing the correct hyperparameter

- Try a bunch of different hyperparameter values
- Fit all of them separately
- See how well each performs
- Choose the best performing one
- It is essential to use cross-validation

Grid search cross-validation

C \ Alpha	0.1	0.2	0.3	0.4
0.5	0.701	0.703	0.697	0.696
0.4	0.699	0.702	0.698	0.702
0.3	0.721	0.726	0.713	0.703
0.2	0.706	0.705	0.704	0.701
0.1	0.698	0.692	0.688	0.675