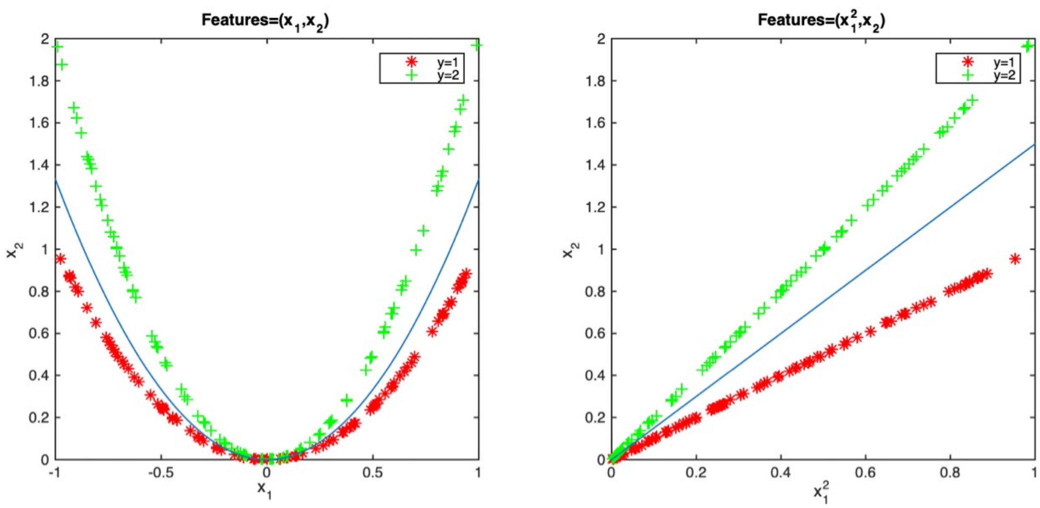
Kernel Trick

Map feature vector to new space

$$\underline{x} \xrightarrow{\underline{\phi}(.)} \underline{\phi}(\underline{x})$$

why?



Dot Products and Kernels

- Building linear classifiers often involves only dot (inner) products between vectors
  - Classifier boundary :
  - Euclidean distance :
- If we map $\underline{x}$ to $\underline{\phi}(\underline{x})$, then dot products in new space formed using

Mercer Kernel If k satisfies the following:

(1) $k(\cdot,\cdot)$ is continuous,

(2) $k(\cdot,\cdot)$ is symmetric, i.e., $k(\underline{x},\underline{v}) = k(\underline{v},\underline{x})$

(3) $k(\cdot,\cdot)$ is positive semi-definite.

Then k is called a Mercer kernel, and for any Mercer kernel it can be shown that there exists

Examples of Kernels

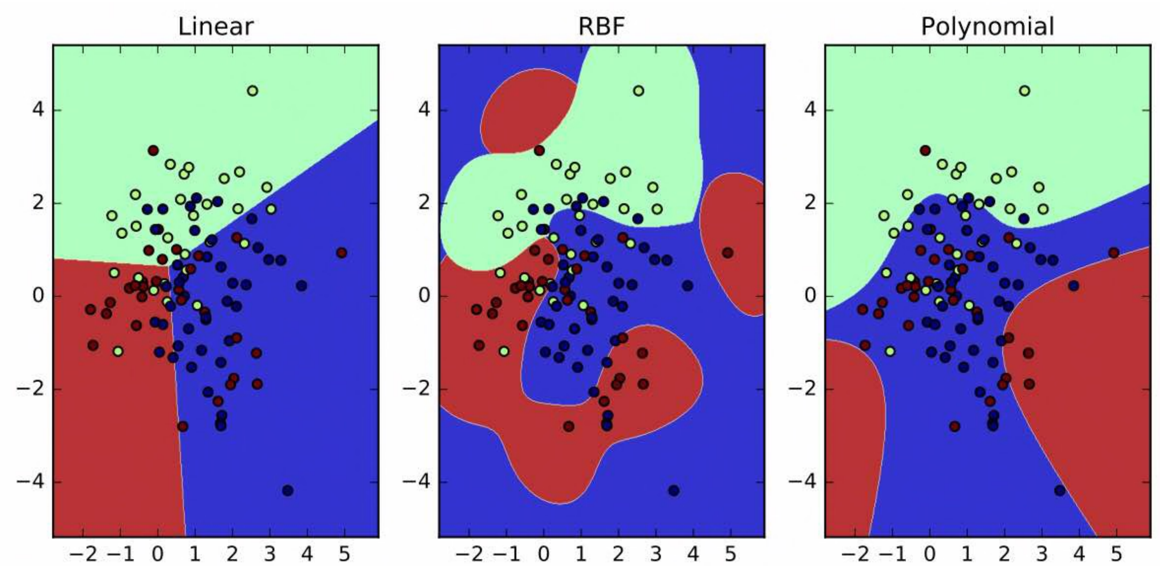
1. Linear: $k(\underline{x},\underline{v}) =$

2. Radial Basis Function (RBF): $k(\underline{x},\underline{v}) =$

e.g. Gaussian RBF: $k(\underline{x},\underline{v}) =$

3. Polynomial kernel: $k(\underline{x},\underline{v}) =$

4. Sigmoid: $k(\underline{x},\underline{v}) =$



SVM

How to Handle Labelled Data

- Need to be careful not to use labelled data in way that leads to
- Classifier that works perfectly on training data, i.e., $Err_{train} = 0$, may perform poorly on actual inputs –

- Two fundamental steps in designing good $f(\cdot)$:

1. Model Selection – Given several models, estimate their performance to choose "best" one
2. Model Assessment Having decided on best model, estimate how well it generalizes.

Split labelled data into 3 parts:

Training	Validation	Test
	Model Selection	Model Assessment

CAUTION Don't touch test data until

Model Selection:

- Use training data on several models; e.g. k-NN, LDA, ..
- Use Validation data to compute Err_{val} for models
- Select best – smallest Err_{val}

Cross-Validation

- In many applications, labelled data
- Cannot afford to split into 3 parts and use exclusively for training, validation, and testing

Step 0 Set aside test data

Step 1 split remaining labelled data into K approximately equal parts $\tau_1, \tau_2, \ldots, \tau_K$

τ_1	τ_2	$\cdots$	τ_K

τ_ℓ — called the

Step 2 For $\ell = 1, 2, \ldots, K$,

- Use all data except that in
- Use

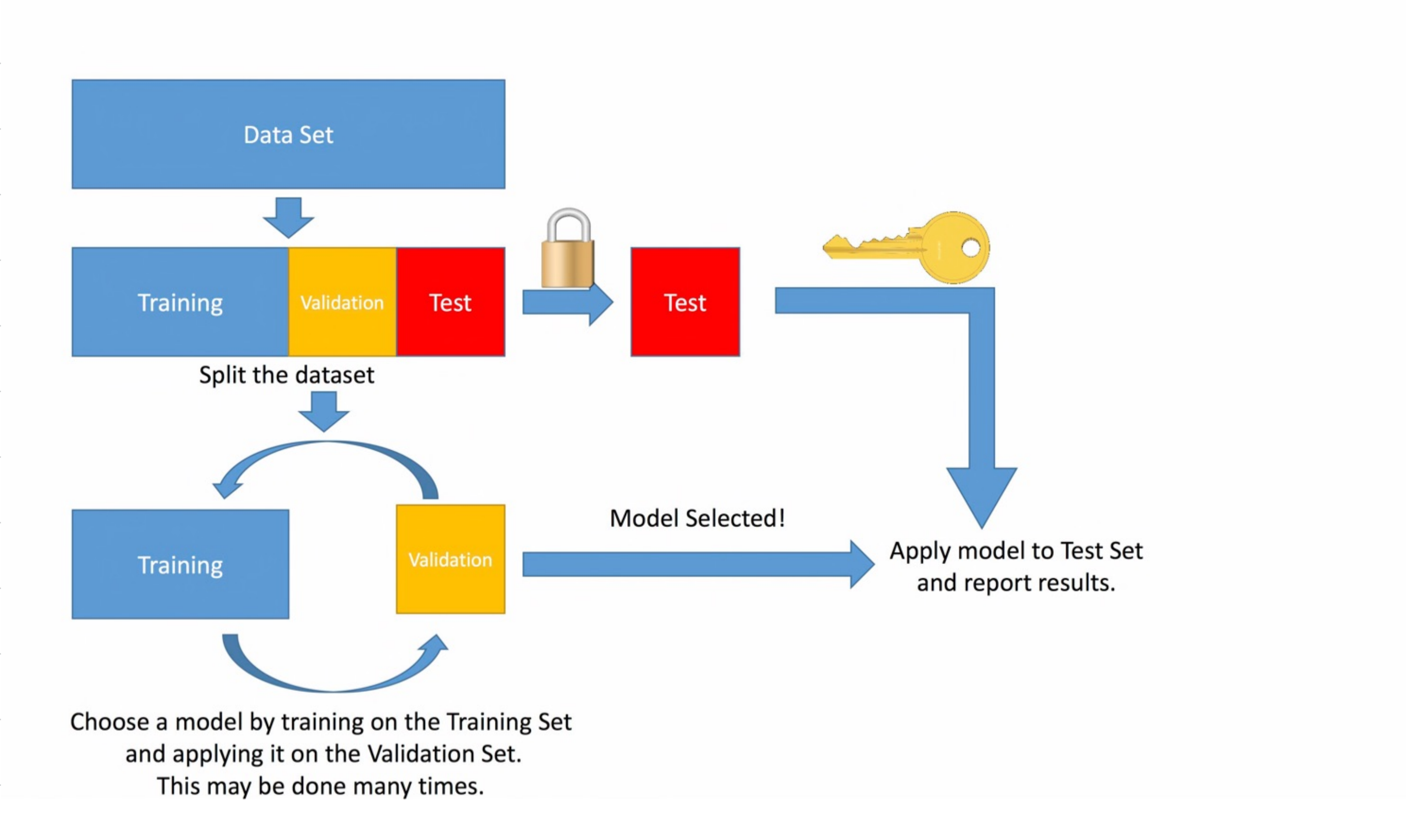
$L_\ell =$ for a given model

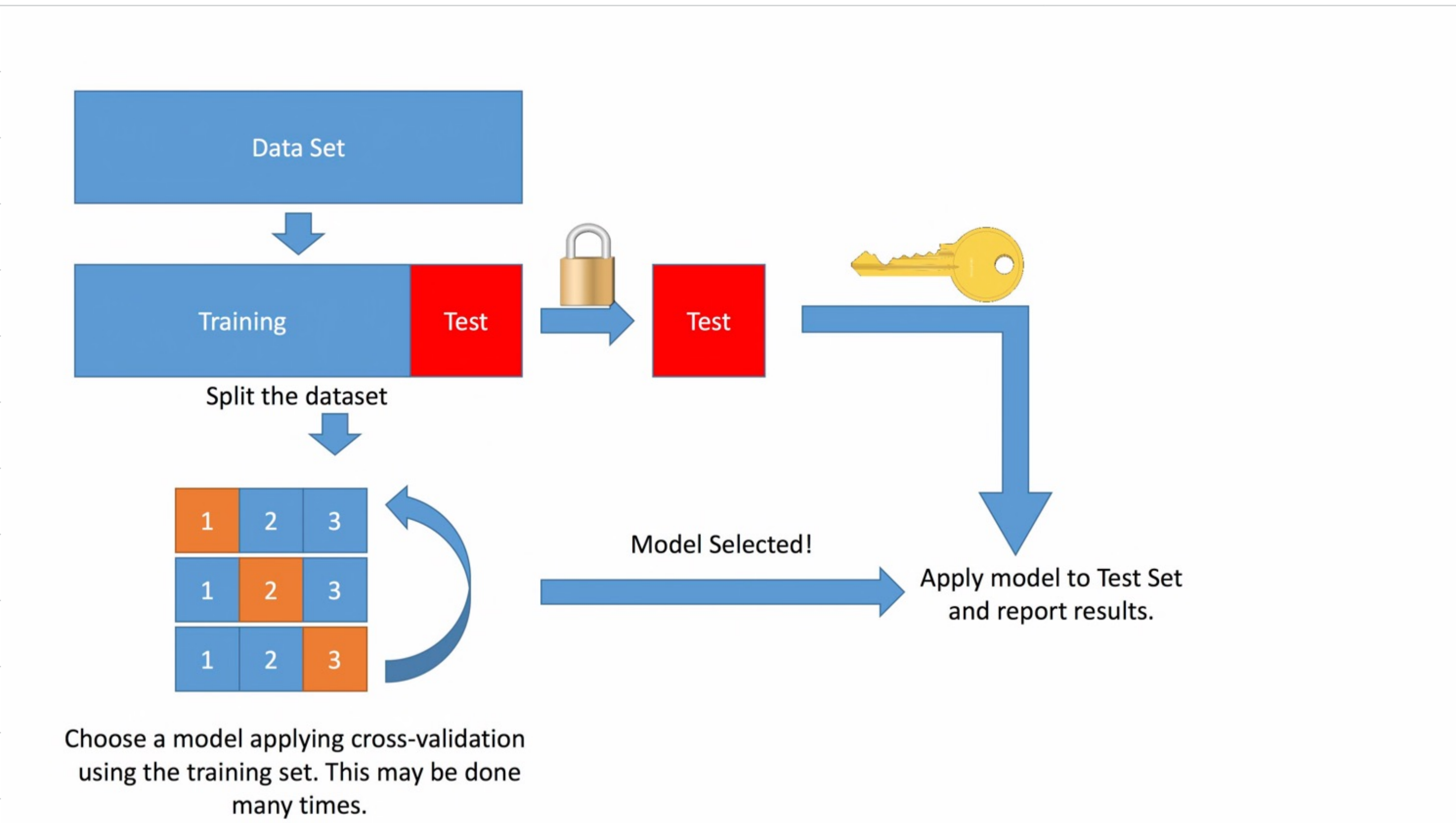
Step 3 Compute cross-validation error:

$$Err_{cv} =$$

Step 4 Use Err_{cv} of different models to

Step 5 Evaluate error of selected model on test data. This gives an accurate estimate of Err_{pred} for selected model


Data Set
Training
Validation
Test
Test
Split the dataset
Training
Validation
Model Selected!
Apply model to Test Set and report results.
Choose a model by training on the Training Set and applying it on the Validation Set. This may be done many times.


Data Set
Training
Test
Test
Split the dataset
1 2 3
1 2 3
1 2 3
Model Selected!
Apply model to Test Set and report results.
Choose a model applying cross-validation using the training set. This may be done many times.