Class core values

1. Be **respect**ful to yourself and others
2. Be **confident** and believe in yourself
3. Always do your **best**
4. Be **cooperative**
5. Be **creative**
6. Have **fun**
7. Be **patient** with yourself while you learn
8. Don’t be shy to **ask “stupid” questions**
9. Be **inclusive** and **accepting**

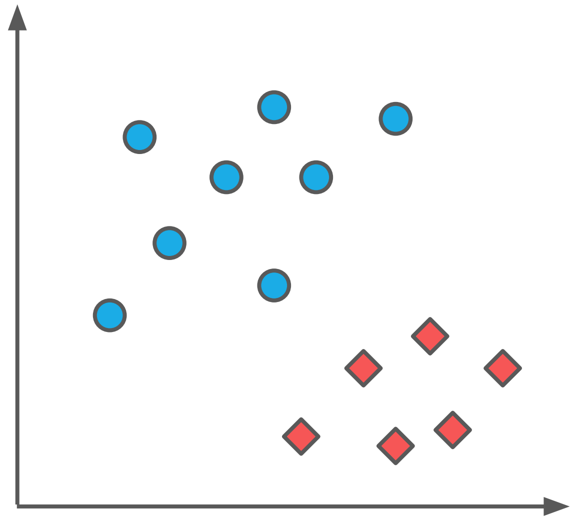
Week 2, Lecture 2

Simple methods can take you a long way

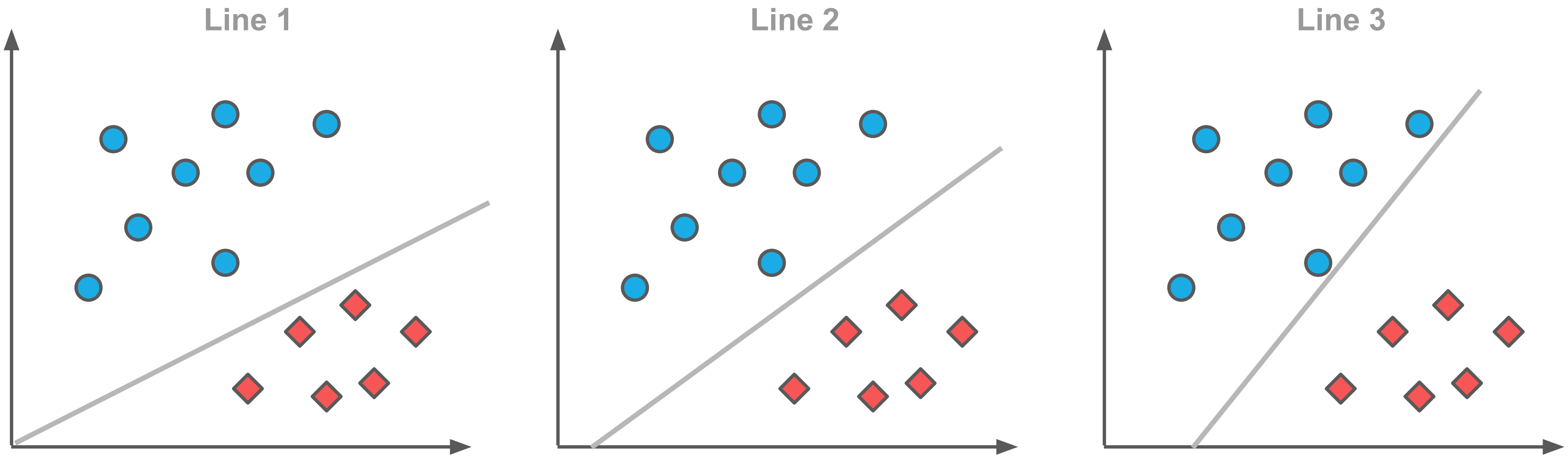
Learning Objectives

1. Describe the basic concept of a support vector machines
2. Describe the basic concept of a kernel
3. Apply computational coding to create and train a SVM
4. Evaluate the performance of models
5. Critically evaluate literature using RF

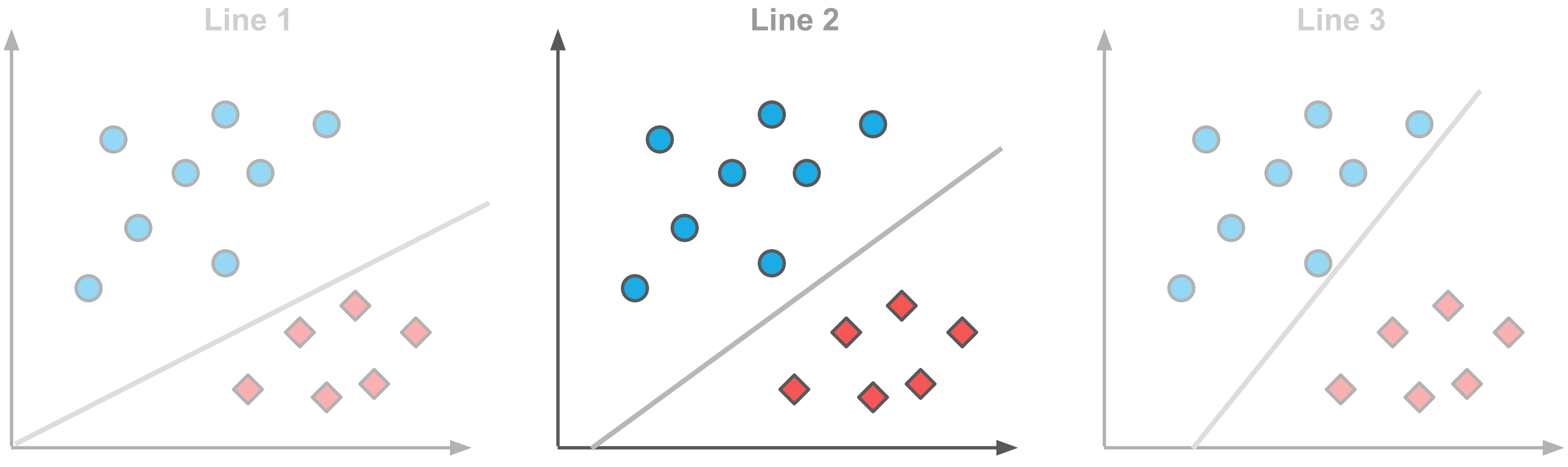
Selecting the best line



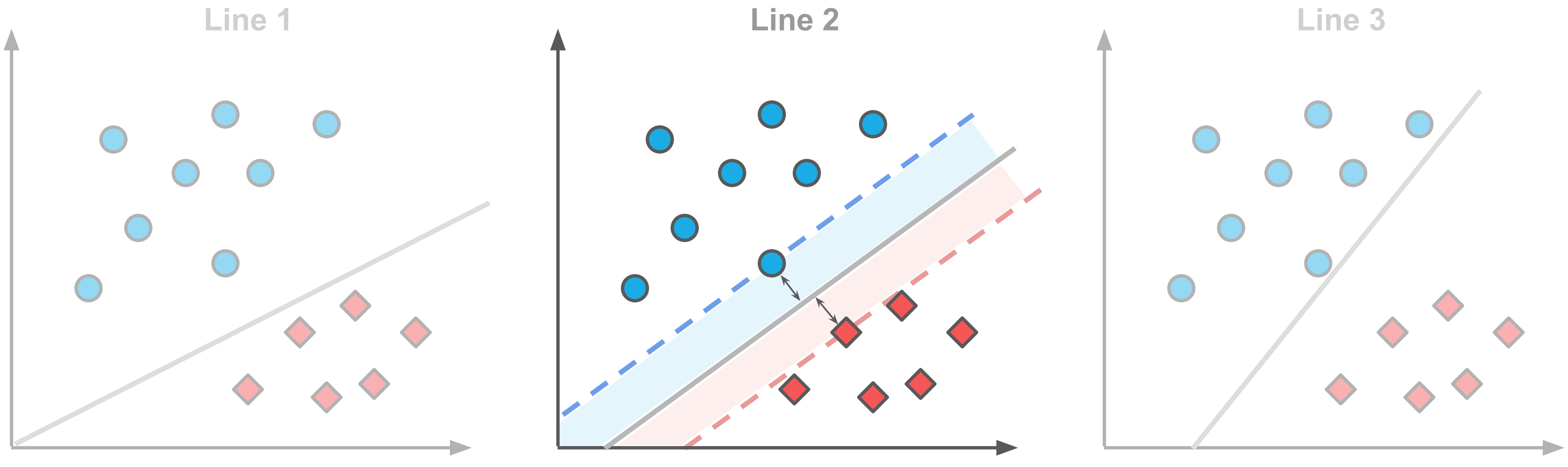
Selecting the best line



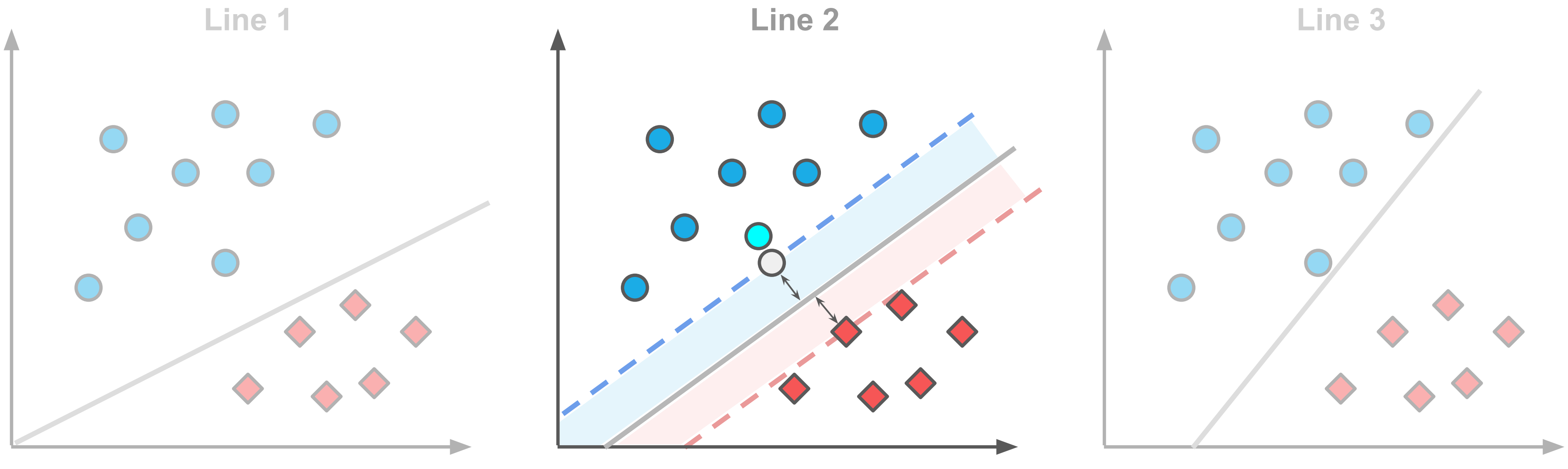
Selecting the best line



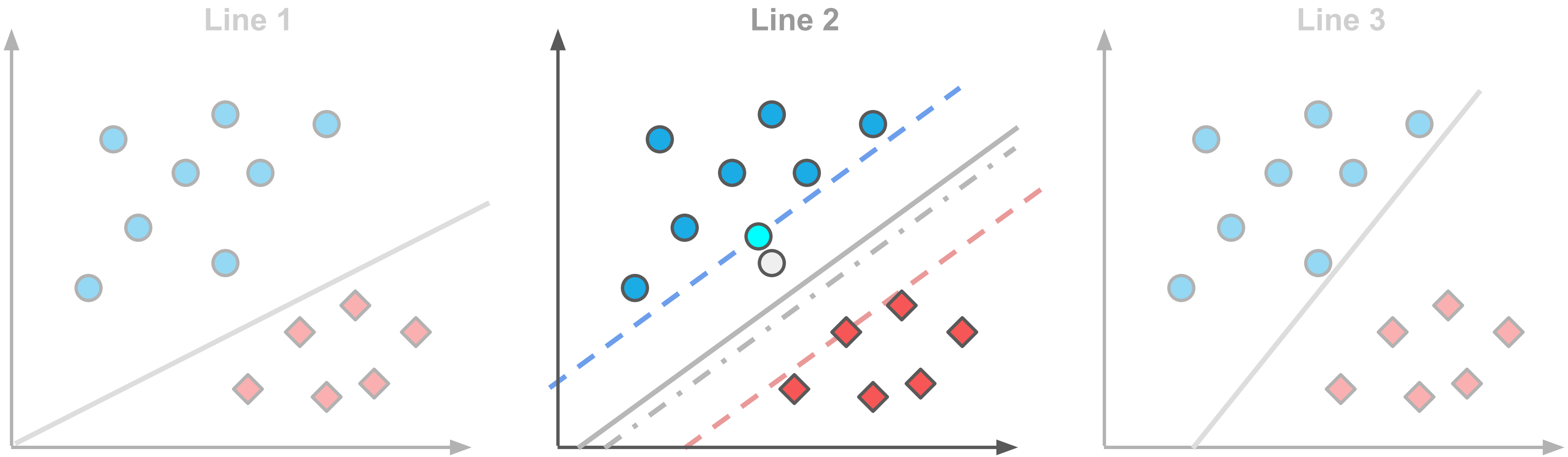
The best separating line is the one that has the largest **margins** from data



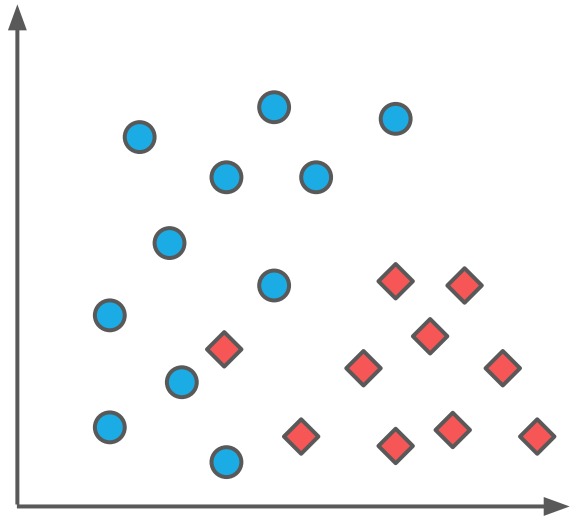
Closest points to the selected lines are called **support vectors**



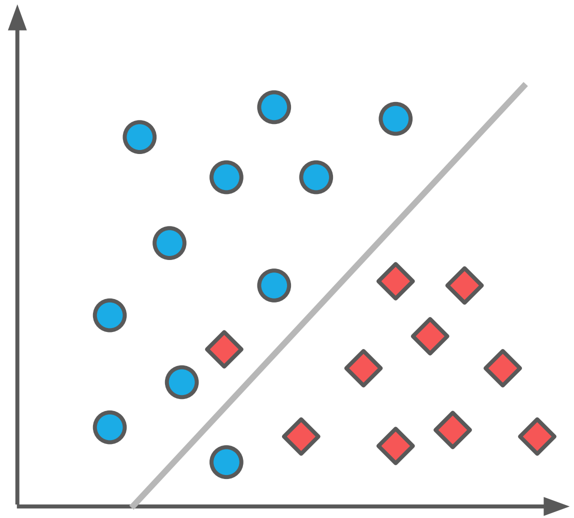
Changing the position of support vectors will change the selection of the best lie



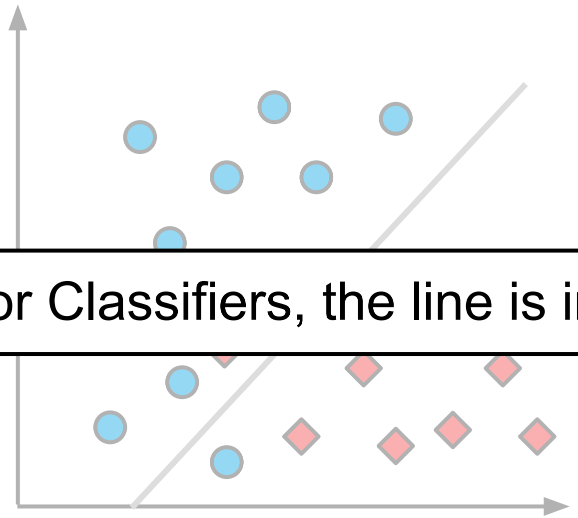
When there is no line that perfectly separates the data …



When there is no line that perfectly separates the data, we choose one that *almost* separates them

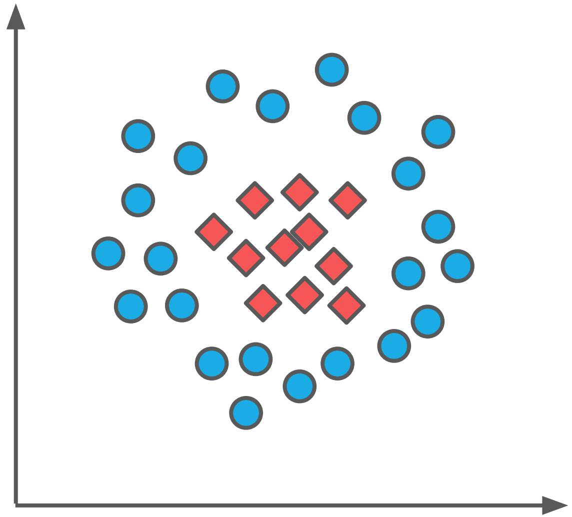


When there is no line that perfectly separates the data, we choose one that *almost* separates them

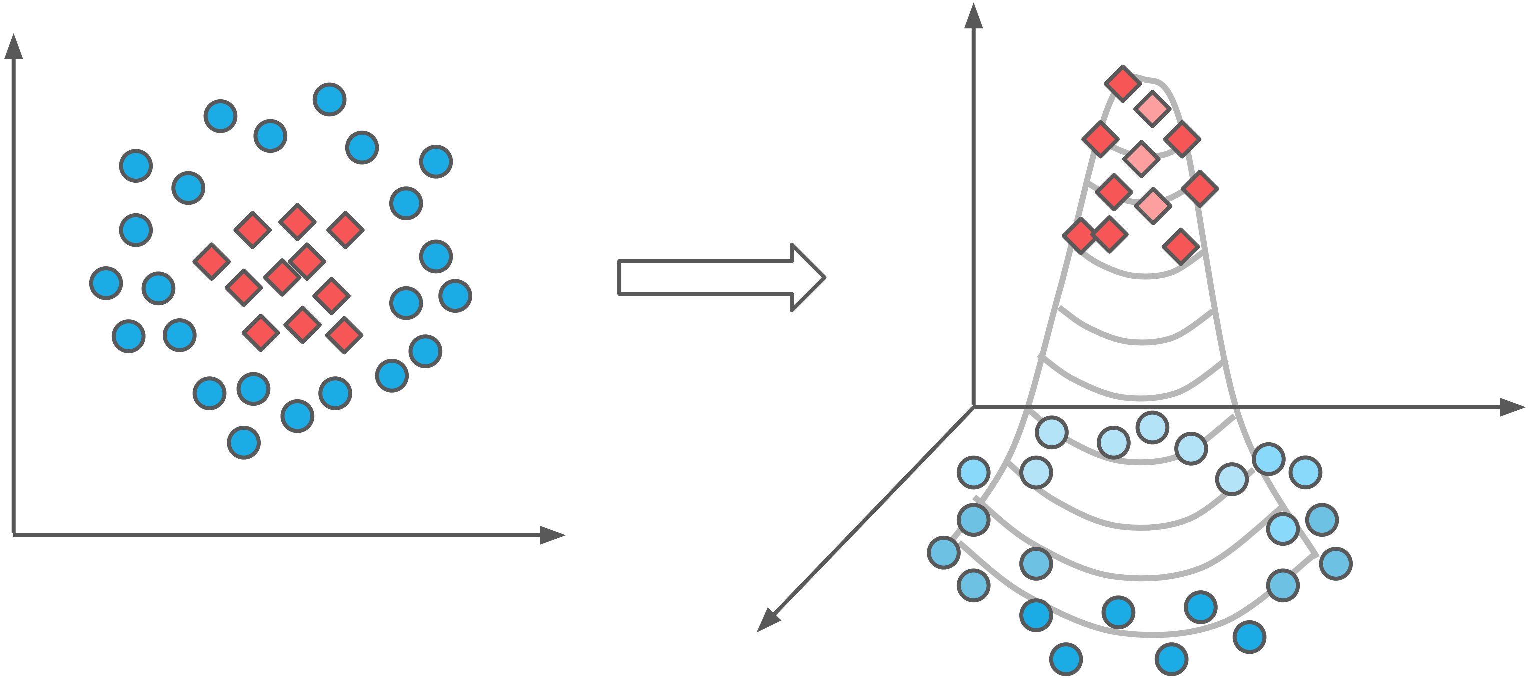


In Support Vector Classifiers, the line is in fact a *hyperplane*

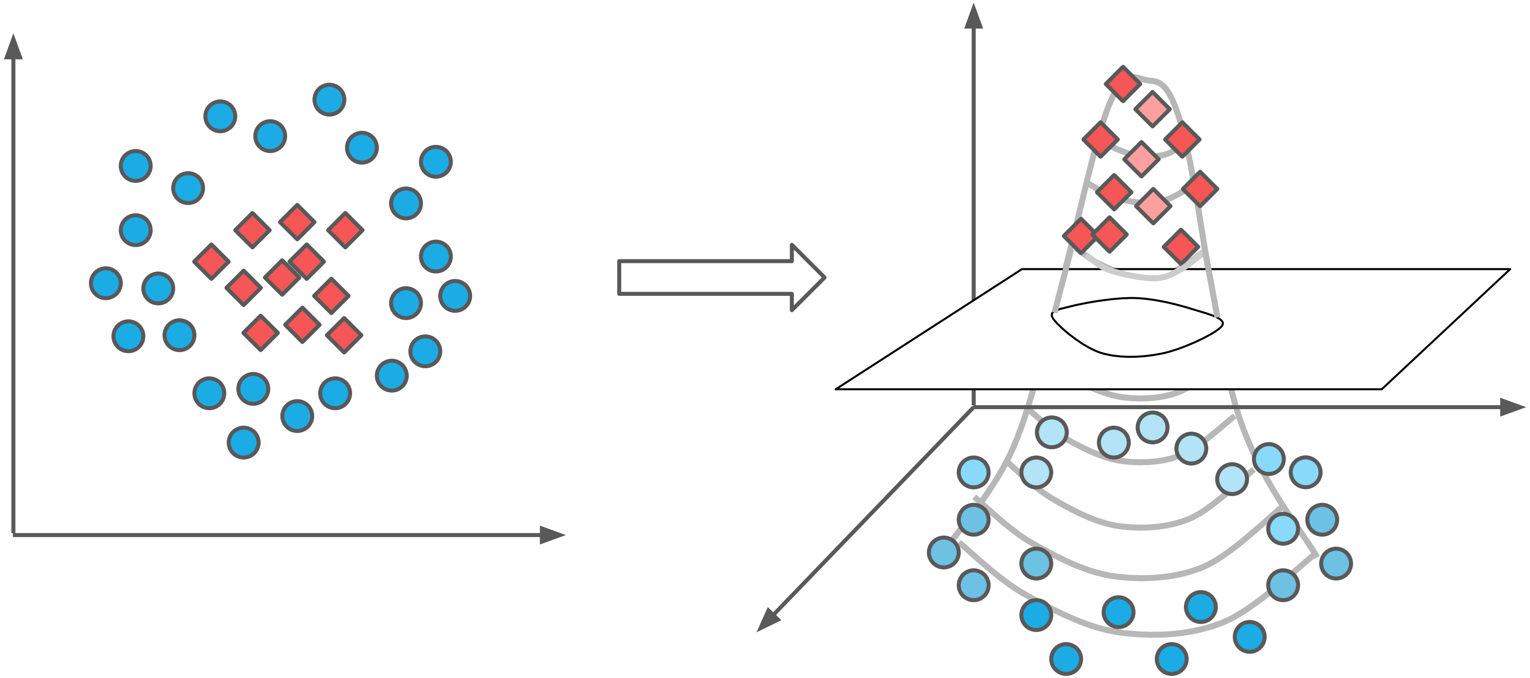
What if there is no linear correlation?



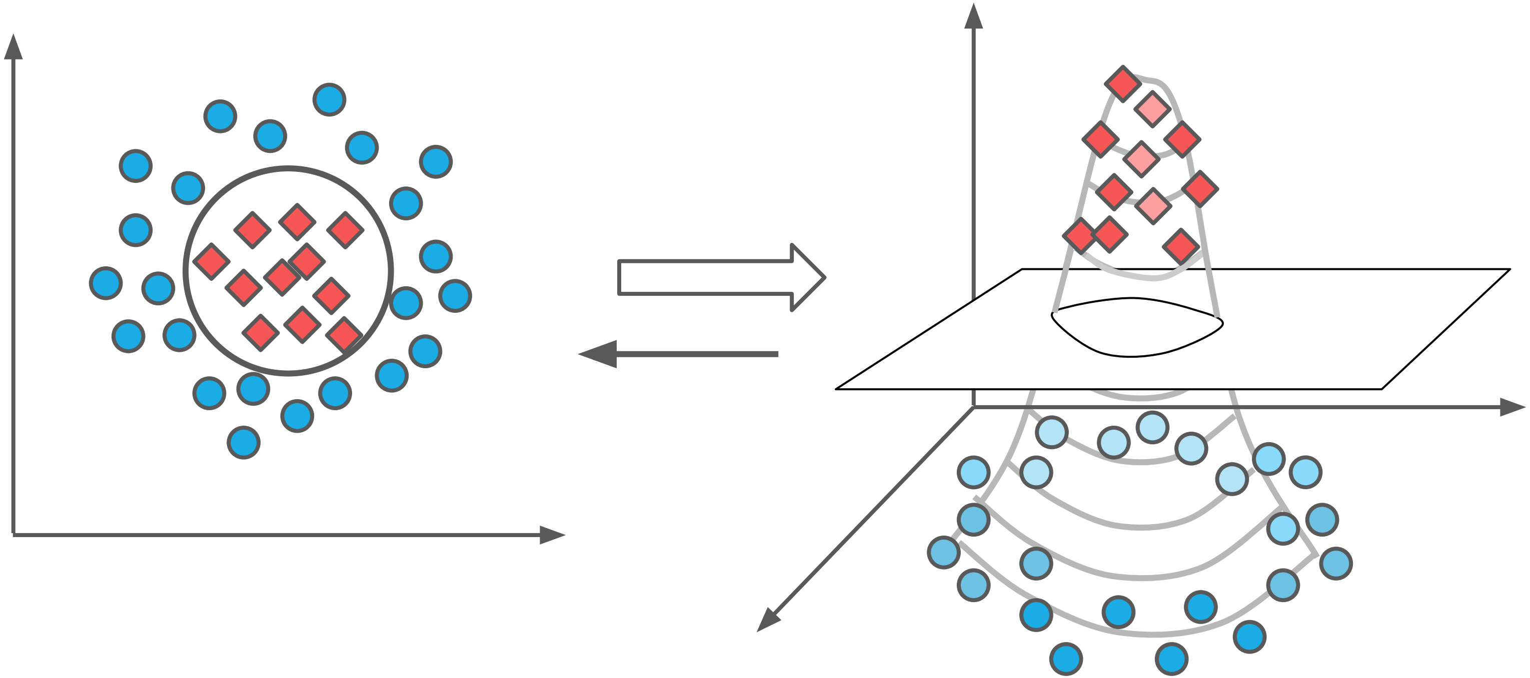
Transforming data into another space can help in finding the desired hyperplane



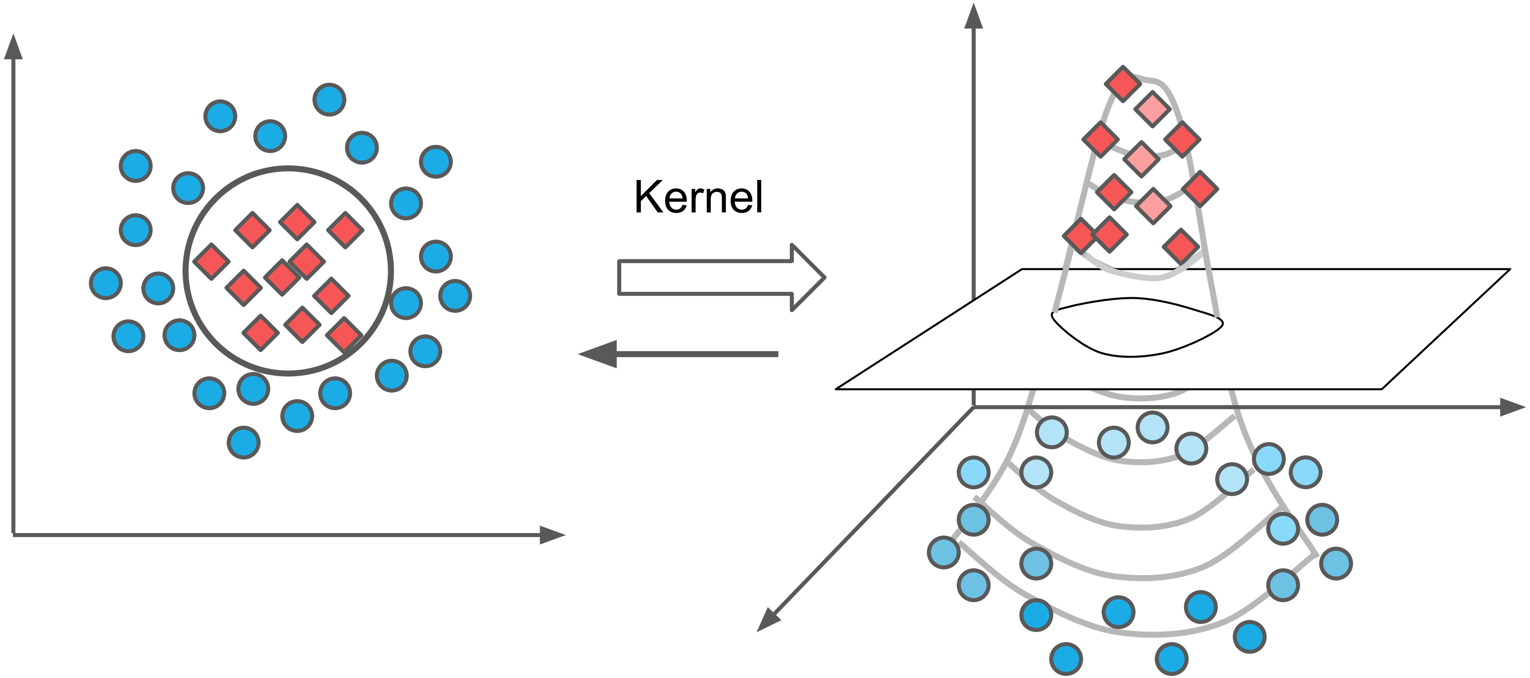
Transforming data into another space can help in finding the desired hyperplane



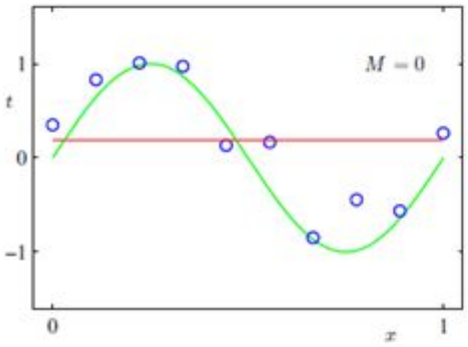
Transforming data into another space can help in finding the desired hyperplane



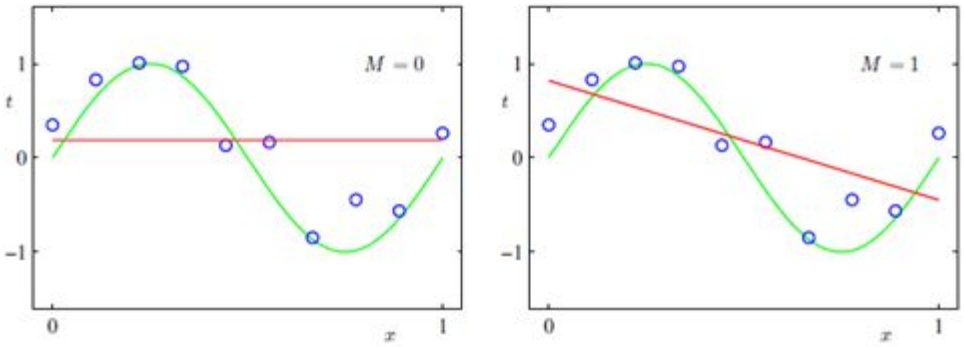
The transformers are called **kernels**



Regularization in a nutshell is a way to avoid overfitting



Regularization in a nutshell is a way to avoid overfitting



Regularization in a nutshell is a way to avoid overfitting

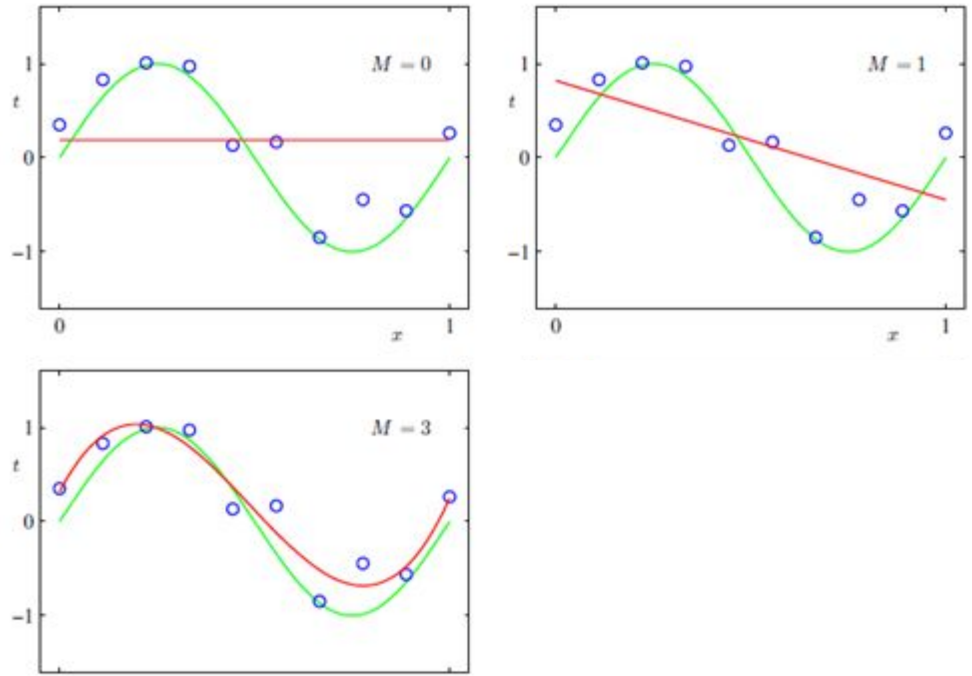


Image from [towards data science: Regularization]

Regularization in a nutshell is a way to avoid overfitting

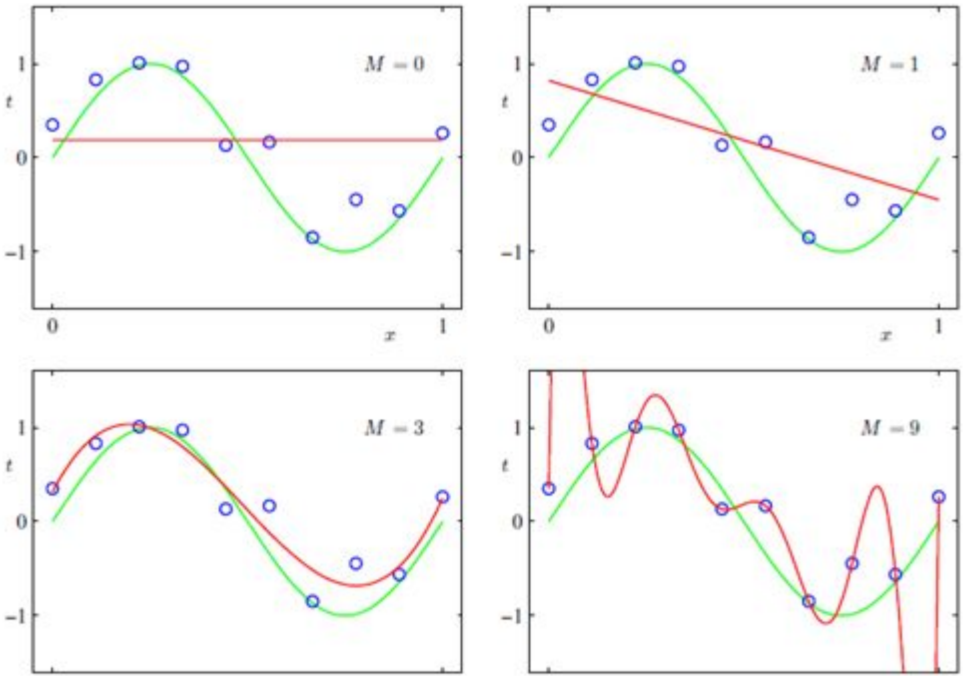
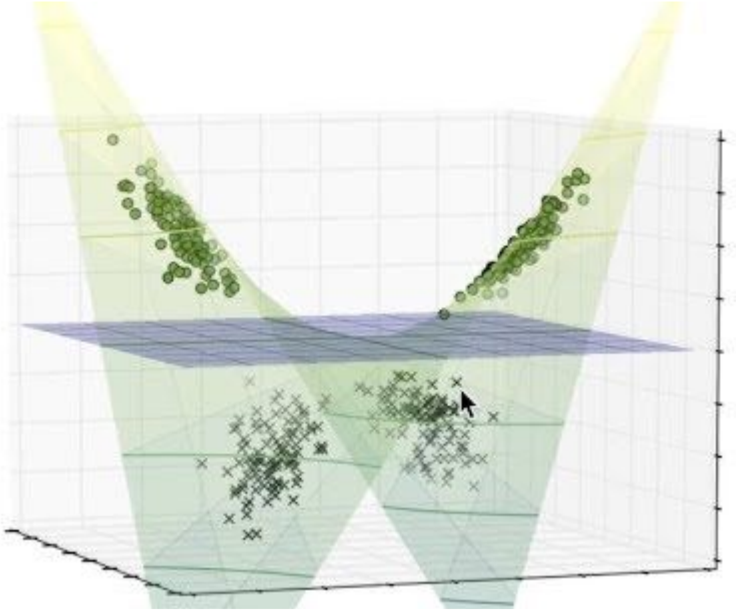


Image from towards data science: Regularization

In-class activity

Applying SVM



Notes on SVM

Pros:

1. Effective in high dimension
2. Works when dimensions (features) are bigger than samples
3. Memory efficient
4. Versatile

Notes on SVM

Pros:

1. Effective in high dimension
2. Works when dimensions (features) are bigger than samples
3. Memory efficient
4. Versatile

Cons:

1. Over-fitting can be a problem when # features >> # samples
2. Do not directly provide probability estimates
3. Not scalable if you have lots of data (> 10000)

Next lecture:

A gentle introduction to a “neuron”

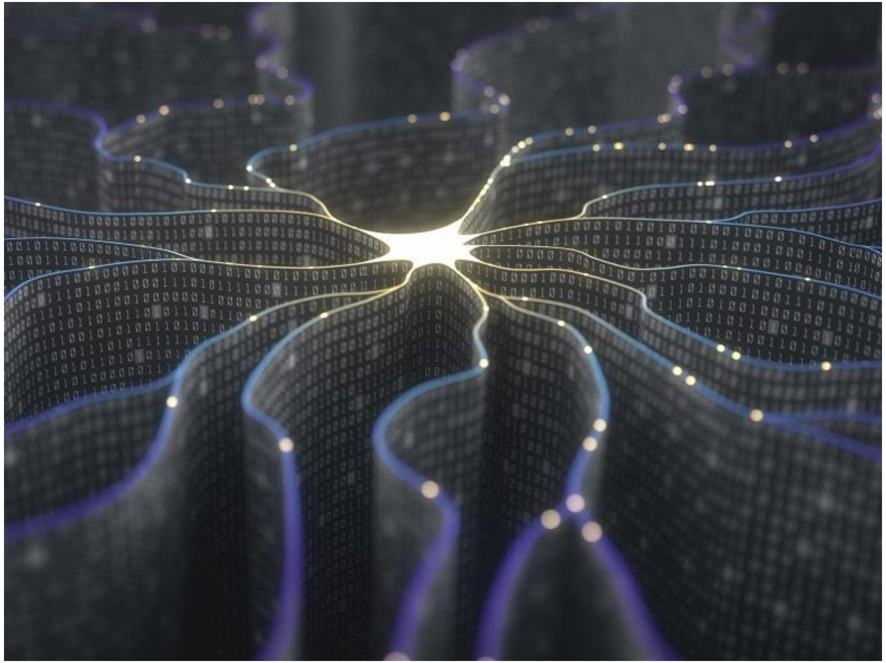


Image from Sci Tech Daily: Deep learning AI explained