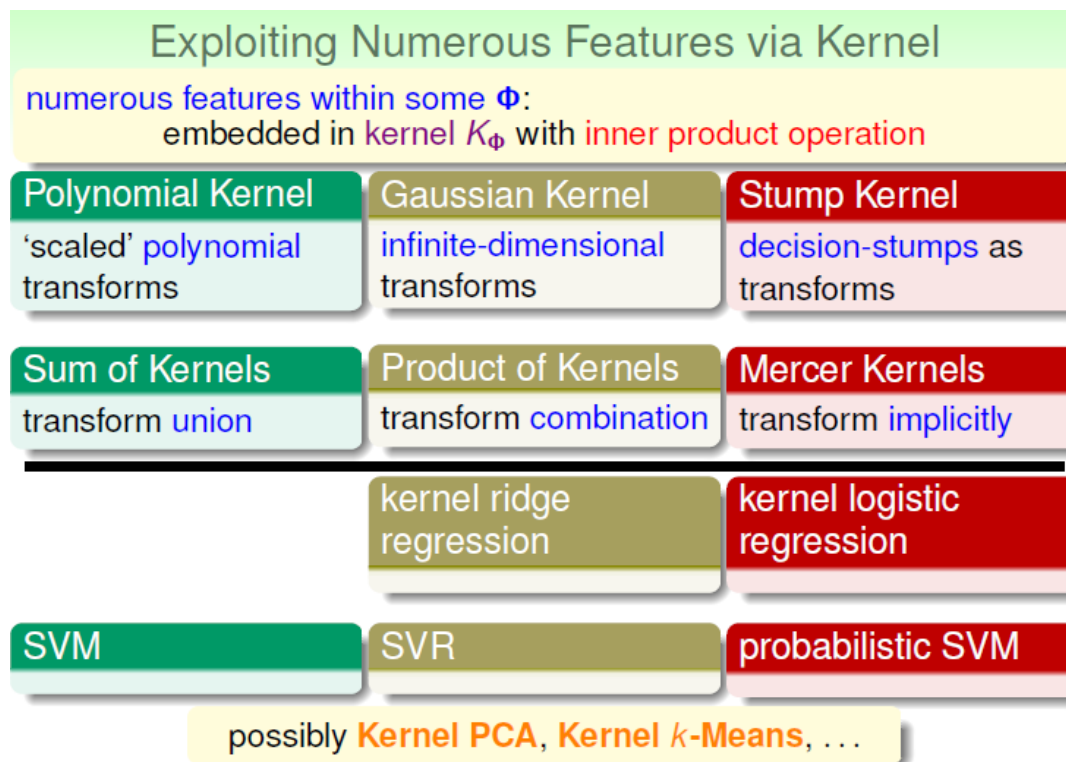


# Finale

收官! 走走停停快一年\*\*emoji 😊 😊 😊

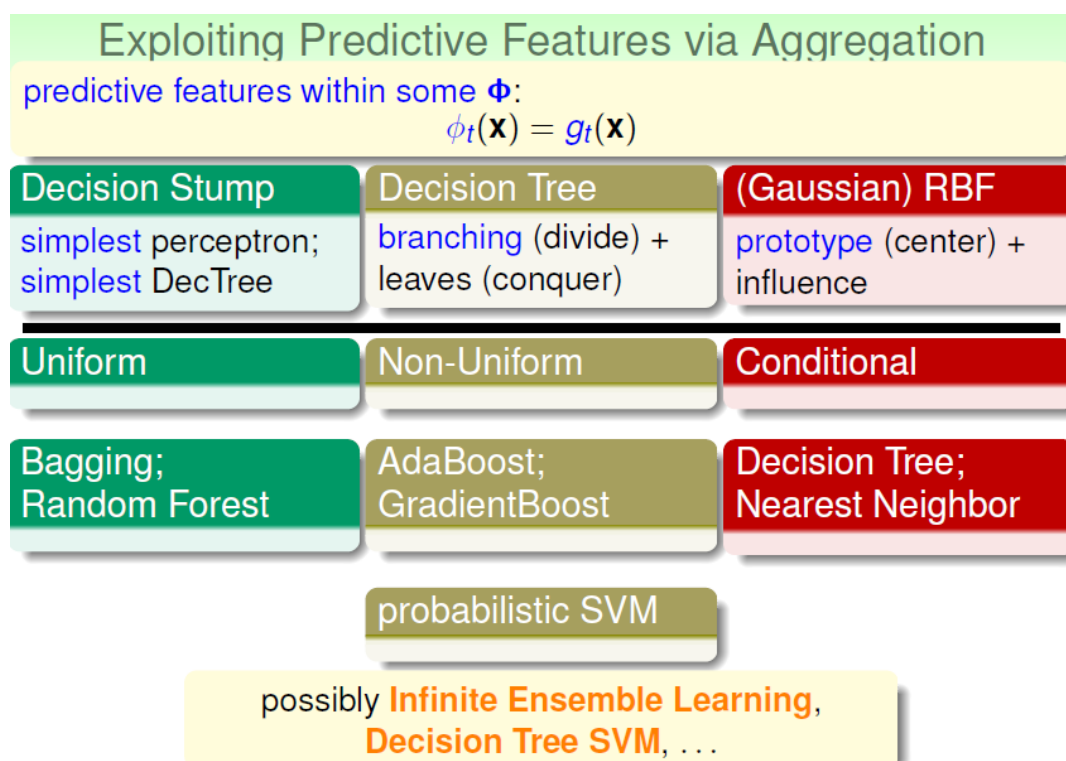
## 一、Feature Exploitation Techniquis

1. kernel family:



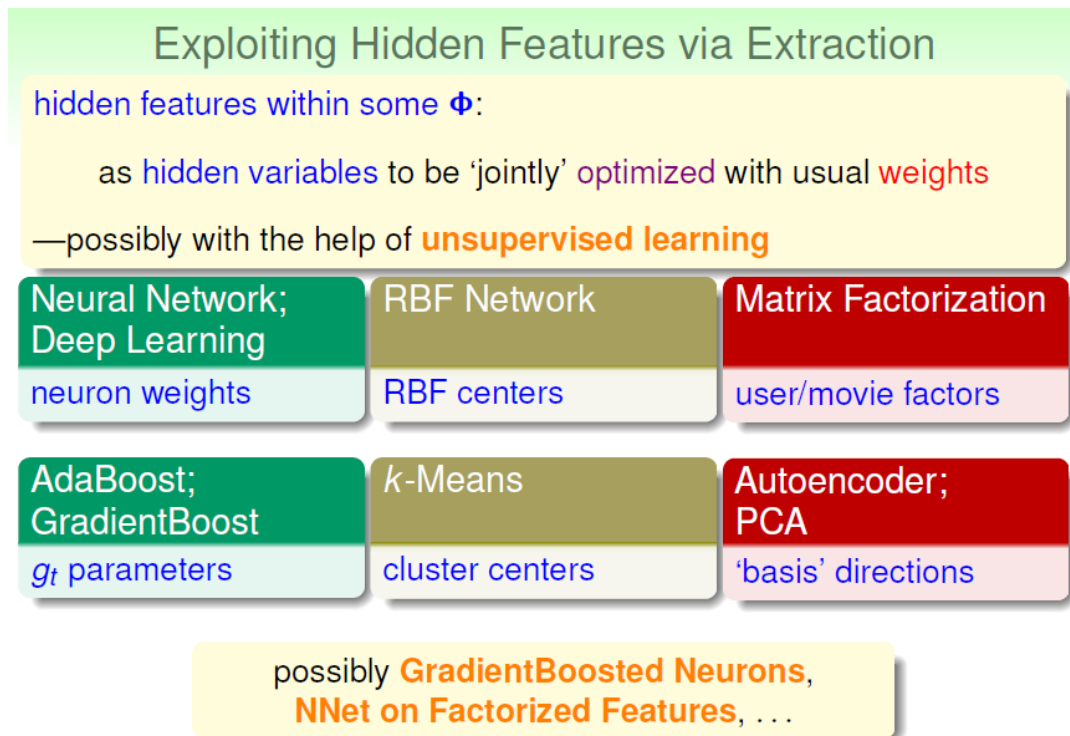
kernel不只适用于SVM

2. aggregation family: ensemble learning集成学习



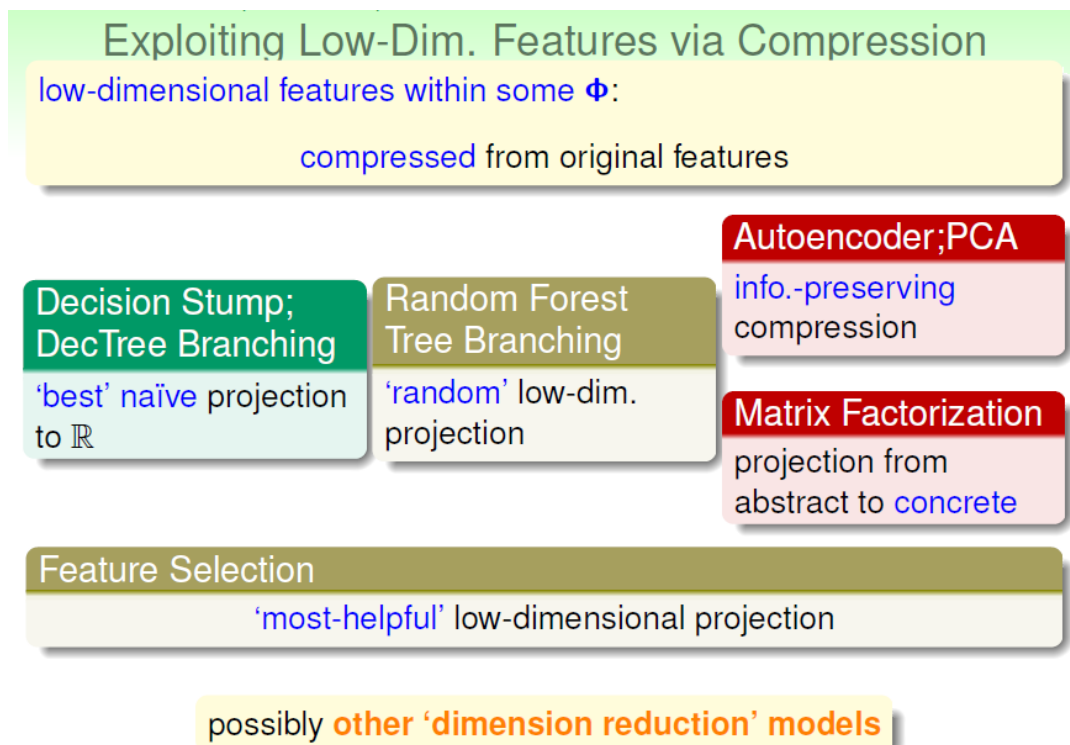
三种机制：民主、加权、条件

3. extraction family：提取特征



非监督学习的方式

4. 压缩降维的方法：把困难的事情简单化



- 我们在decision tree里面用decision stump;
- 我们在random forest中的随机投影;
- 我们在auto encoder、pca中直接的降维;
- 我们在matrix factorization中直接分解分析eigenvalue

## 二、Error Optimization Techniques

1. GD family: first-order optimization

## Numerical Optimization via Gradient Descent

when  $\nabla E$  'approximately' defined, use it for **1st order approximation**:

$$\text{new variables} = \text{old variables} - \eta \nabla E$$

### SGD/Minibatch/GD

(Kernel) LogReg;  
Neural Network  
[backprop];  
Matrix Factorization;  
Linear SVM (maybe)

### Steepest Descent

AdaBoost;  
GradientBoost

### Functional GD

AdaBoost;  
GradientBoost

possibly **2nd order techniques**,  
**GD under constraints**, ...

还有牛顿法这些二阶优化

2. 化归思想的应用——转换问题:

## Indirect Optimization via Equivalent Solution

when difficult to solve original problem,  
seek for **equivalent solution**

### Dual SVM

equivalence via  
**convex QP**

Kernel LogReg  
Kernel RidgeReg

equivalence via  
**representer**

### PCA

equivalence to  
**eigenproblem**

some **other boosting models** and **modern solvers of kernel models** rely on such a technique heavily

- 对偶SVM: 凸二次优化问题
- kernel logReg和RidgeReg: 表示定理
- PCA: 特征值问题

3. 多步学习策略:

## Complicated Optimization via Multiple Steps

when difficult to solve original problem,  
seek for 'easier' sub-problems

### Multi-Stage

probabilistic SVM;  
linear blending;  
stacking;  
RBF Network;  
DeepNet pre-training

### Alternating Optim.

k-Means;  
alternating LeastSqr;  
(steepest descent)

### Divide & Conquer

decision tree;

useful for complicated models

- 多步骤学习: blending、learning、stacking等等
- 交互式学习: K-Means、alternating LeastSqr(最陡梯度)
- 分治法学习: decision tree

## 三、Overfitting Elimination Techniques

1. 正则化:

### Overfitting Elimination via Regularization

when model too 'powerful':

add brakes somewhere

### large-margin

SVM;  
AdaBoost (indirectly)

### L2

SVR;  
kernel models;  
NNet [weight-decay]

### voting/averaging

uniform blending;  
Bagging;  
Random Forest

### denoising

autoencoder

### weight-elimination

NNet

### constraining

autoenc. [weights];  
RBF [# centers];

### pruning

decision tree

### early stopping

NNet (any GD-like)

arguably most important techniques

随时随地踩刹车! 这比开车重要

实际上正则化就是降低模型复杂度来让泛化能力提高, 着重考虑的是结构风险最优化。

2. 验证集:

## Overfitting Elimination via Validation

when model too 'powerful':

check performance carefully and honestly

# SV

SVM/SVR

OOB

Random Forest

Internal Validation

blending;

DecTree pruning

simple but **necessary**

作为模型选择的依据

## 四、Machine Learning in Action

machine learning jungle!

### Machine Learning Jungle

bagging   decision tree   support vector machine   neural network   kernel  
AdaBoost   aggregation   sparsity   autoencoder   functional gradient  
dual   uniform blending   deep learning   nearest neighbor   decision stump  
kernel LogReg   large-margin   prototype   quadratic programming   SVR  
GBDT   PCA   random forest   matrix factorization   Gaussian kernel  
soft-margin   k-means   OOB error   RBF network   probabilistic SVM

welcome to the **jungle**!

平凡生活的英雄主义梦想!