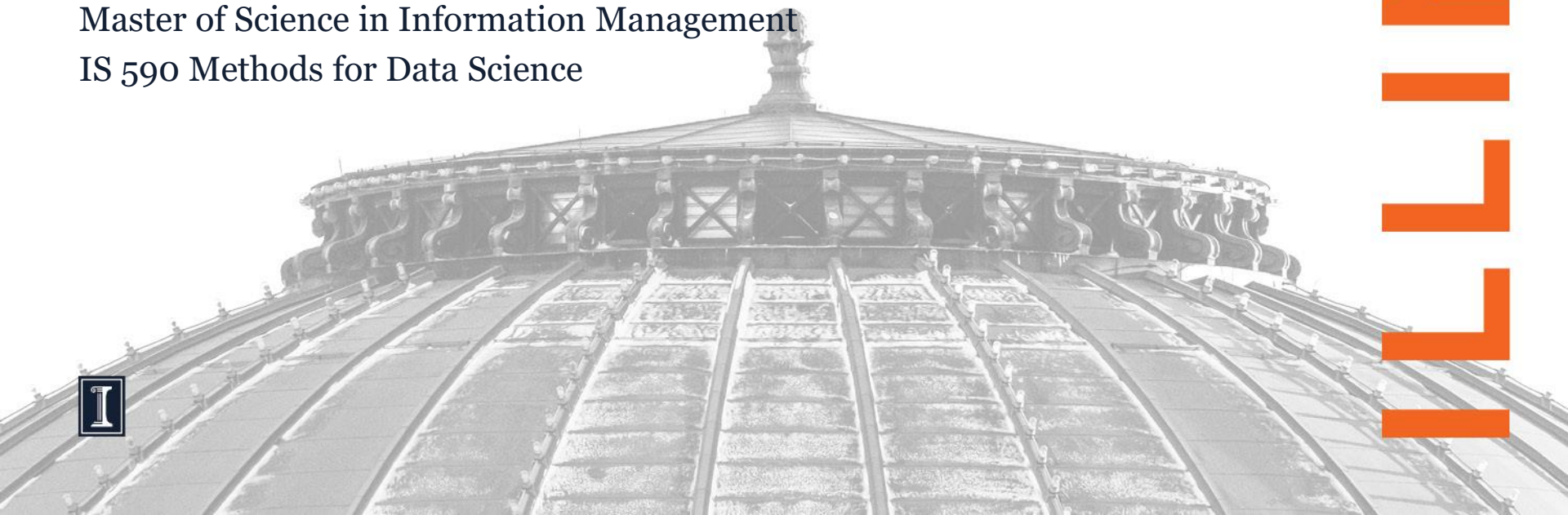
Real Estate Profitability

An Analysis of the Residential Building Dataset with the Support Vector Machine Method

Te Lin
Master of Science in Information Management
IS 590 Methods for Data Science



ILLINOIS

Agenda

- Background
- Dataset Overview
- Goal
- Support Vector Machine
  - Radial Kernel
  - Polynomial Kernel
- Diagnostics

Background

- What?
  - Residential buildings
- Where?
  - Tehran, Iran
- When?
  - From 1972 to 1990
- Source
  - Machine Learning Repository hosted by University of California, Irvine

Photograph via loki010 on Reddit

Dataset Overview

Variable Group	Variable ID	Descriptions	Unit	Time Lag Number [p]
PROJECT PHYSICAL AND FINANCIAL VARIABLES	V-1	Project locality defined in terms of zip codes	N/A	N/A
	V-2	Total floor area of the building	m^2	N/A
	V-3	Lot area	m^2	N/A
	V-4	Total preliminary estimated construction cost based on the prices at the beginning of the project	10000000 IRR^m	N/A
	V-5	Preliminary estimated construction cost based on the prices at the beginning of the project	10000 IRR^m	N/A
	V-6	Equivalent preliminary estimated construction cost based on the prices at the beginning of the project in a selected base year [a]	10000 IRR^m	N/A
	V-7	Duration of construction	As a number of time resolution [e]	N/A
	V-8	Price of the unit at the beginning of the project per m^2	10000 IRR^m	N/A
	V-9	Actual sales prices (output)	10000 IRR^m	N/A
	V-10	Actual construction costs (output)	10000 IRR^m	N/A
ECONOMIC VARIABLES AND INDICES [n]	V-11	The number of building permits issued	N/A	1 to 5
	V-12	Building services index (BSI) [b] for a preselected base year [a]	N/A	1 to 5
	V-13	Wholesale price index (WPI) [c] of building materials for the base year	N/A	1 to 5
	V-14	Total floor areas of building permits issued by the city/municipality	m^2	1 to 5
	V-15	Cumulative liquidity [d]	10000000 IRR^m	1 to 5
	V-16	Private sector investment in new buildings	10000000 IRR^m	1 to 5
	V-17	Land price index for the base year [a]	10000000 IRR^m	1 to 5
	V-18	The number of loans extended by banks in a time resolution [e]	N/A	1 to 5
	V-19	The amount of loans extended by banks in a time resolution [e]	10000000 IRR^m	1 to 5
	V-20	The interest rate for loan in a time resolution [e]	%	1 to 5
	V-21	The average construction cost of buildings by private sector at the time of completion of construction	10000 IRR^m /m^2	1 to 5
	V-22	The average of construction cost of buildings by private sector at the beginning of the construction	10000 IRR^m /m^2	1 to 5
	V-23	Official exchange rate with respect to dollars	IRR^m	1 to 5
	V-24	Nonofficial (street market) exchange rate with respect to dollars [h]	IRR^m	1 to 5
	V-25	Consumer price index (CPI) [i] in the base year [a]	N/A	1 to 5
	V-26	CPI of housing, water, fuel & power in the base year [a]	N/A	1 to 5
	V-27	Stock market index [j]	N/A	1 to 5
	V-28	Population of the city	N/A	1 to 5
	V-29	Gold price per ounce	IRR^m	1 to 5

Dataset Overview

- Response (1):
  - Profitability (binary)
  - Ratio = Actual sales prices/Actual construction costs
  - Levels:
    - “Y” if ratio > 5
    - “N” if ratio < 5

- Predictors (27):
  - 8 project physicals and financial variables (V-1 to V- 8)
  - 19 economic variables and indices (V-11 to V-29)

Goal

- Build a SVM with either a polynomial kernel or a radial kernel
- Ys on a side of a kernel, while Ns on the other side of a kernel.
- Make inference if the profitability of a residential building is larger than a particular threshold given physical, financial and economic variables.

Why Support Vector Machine

- Dataset
  - Categorical response and continuous predictors
- Robust
  - Only support vectors would affect kernels
  - Resistant to outliers
- Options
  - Combines support vector classifier with a non-linear kernel
  - Radial or polynomial

SVM – Radial Kernel

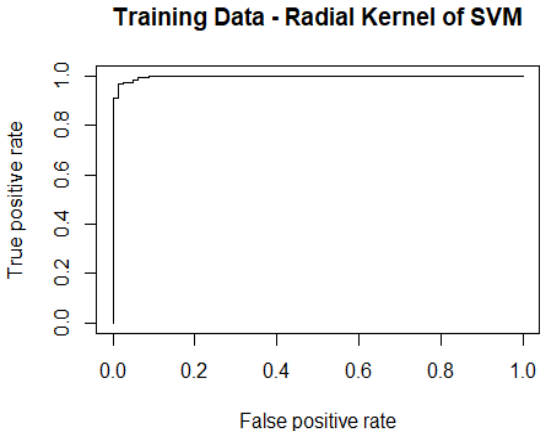
- Select the best choice of gamma and cost for an SVM with radial kernel

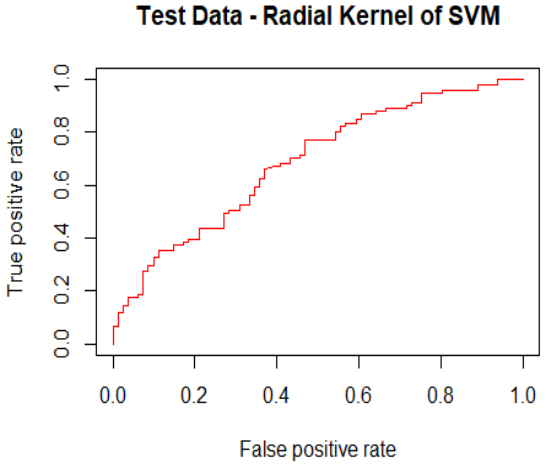
```
tune.out = tune(svm, prof~., data = mydata[train, ], kernel ='radial',
                ranges =
                  list(cost = c(0.1, 1, 10, 100, 1000),
                       gamma = c(0.5, 1, 2, 3, 4)),
                decision.values = TRUE)
summary(tune.out)
## Parameter tuning of 'svm':
## - sampling method: 10-fold cross validation
##
## - best parameters:
##  cost gamma
##     1   0.5
## - best performance: 0.295
```

SVM – Radial Kernel

```
summary(bestmod.radial)
## Call:
## best.tune(method = svm, train.x = prof ~ ., data = mydata[train,
##     ], ranges = list(cost = c(0.1, 1, 10, 100, 1000), gamma = c(0.5,
##     1, 2, 3, 4)), kernel = "radial", decision.values = TRUE)
##
## Parameters:
##    SVM-Type:  C-classification
##  SVM-Kernel:  radial
##        cost:  1
##       gamma:  0.5
##
## Number of Support Vectors:  193
##
##  ( 115 78 )
##
## Number of Classes:  2
## Levels:
##  N Y
```

		Truth	
Predict		N	Y
	N	27	10
	Y	54	81
	Error Rate: 0.372		





SVM – Polynomial Kernel

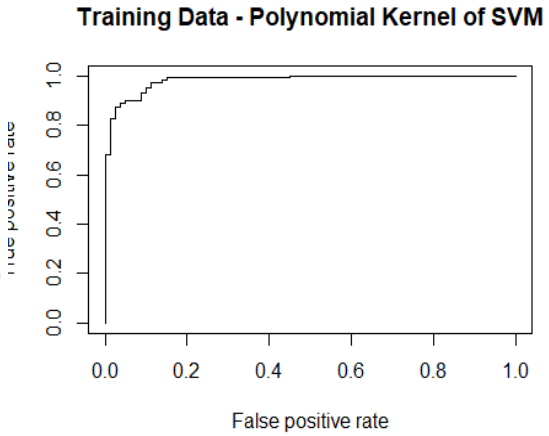
- Select the best choice of degree and cost for an SVM w ith polynomial kernel

```
tune.out = tune(svm, prof~., data = mydata[train,], kernel = 'polynomial',
                ranges =
                  list(degree =c(1, 2, 3, 4, 5),
                       cost = c(0.1, 1, 10, 100, 1000)),
                decision.values = TRUE)
summary(tune.out)
## Parameter tuning of 'svm':
## - sampling method: 10-fold cross validation
## - best parameters:
##  degree cost
##       1  100
## - best performance: 0.115
```

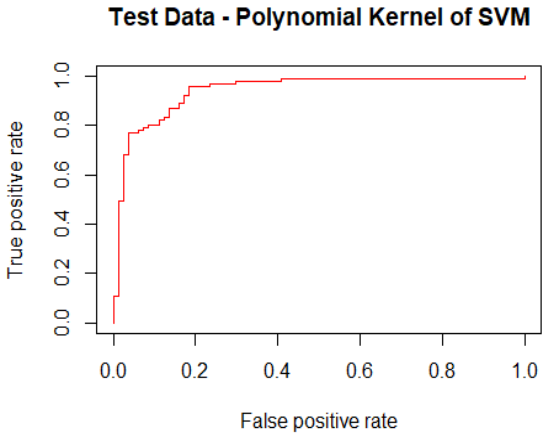
Support vector classifier with a linear decision boundary

SVM – Polynomial Kernel

```
summary(bestmod.poly)
##
## Call:
## best.tune(method = svm, train.x = prof ~ ., data = mydata[train,
##     ], ranges = list(degree = c(1, 2, 3, 4, 5), cost = c(0.1,
##     1, 10, 100, 1000)), kernel = "polynomial", decision.values = TRUE)
##
## Parameters:
##    SVM-Type:  C-classification
##  SVM-Kernel:  polynomial
##        cost:  100
##      degree:  1
##       gamma:  0.03703704
##      coef.0:  0
## Number of Support Vectors:  63
##  ( 30 33 )
##
## Number of Classes:  2
##
## Levels:
##  N Y
```



		Truth	
Predict		N	Y
	N	66	7
	Y	15	84
	Error Rate: 0.128		



Diagnostics

- Bias-variance tradeoff

Cost	Margin	Bias	Variance
decreases	wide	low	high
increases	narrow	high	low

- A more flexible method will often produce a lower training error rate, but does not necessarily lead to improved performance on test data
- The choice of tuning parameter determine the extent to which the model underfits or overfits the data

Reference

- Rafiei, M.H. and Adeli, H. (2015). A Novel Machine Learning Model for Estimation of Sale Prices of Real Estate Units. ASCE, Journal of Construction Engineering & Management, 142(2), 04015066.