COURSE MEDIAN PREDICTION VIA SYLLABI ANALYSIS

Coralie Phanord
Graeson McMahon
Kelsey Justis

Milestone Presentation
Cosc 74: Machine Learning & Statistical Analysis

DATA COLLECTION/FORMATTING

- Collected syllabi through Dartmouth websites and department heads.
  - ~ 500 syllabi PDFs collected
  - Of these, ~35-40% unusable (median not published OR unparsable)
- Used Xpdf's *pdftotext* application to convert syllabi to parsable .txt files
- Wrote code to pair each course with its median based on the name of its respective .txt file
  - Error prone; requires manual work when no match found. As a result, only 235 syllabi used in test.

TEXT PARSING

COSC-70-14S.txt — Open with TextEdit

CS070/CS170, Spring 2013
Numerical and Computational Tools for Applied Science
About Syllabus
Course description
This course provides a practical and principled coverage of numerical and computational tools of use in many scientific disciplines. The focus is on the analysis and application of numerical methods for linear algebra, optimization, and function approximation. The course also provides an introduction to Matlab, a programming environment for scientific computing. This course is designed for undergraduate and graduate students across the sciences and social sciences.
Administrative information
Instructor Gevorg Grigoryan | 113 Sudikoff | office hours: by appointment
Lectures T/Th 2:00 pm 3:50 pm | xhour (may be used occasionally to make up for cancelled classes) W 4:15 5:05
Lab Sudikoff 001: Linux machines with Matlab. As an alternative, you can install and use Matlab on your machine by following the instructions provided here.
Textbooks (these books are suggested as additional references; they are not required) Gilbert Strang, Linear Algebra and Its Applications (4th Edition), Brooks Cole 2005 Michael T. Heath, Scientific Computing: An Introductory Survey, McGrawHill 2002
Grading and policies
Grading scheme Course grades will be based on four homework assignments (60%), final project (30%) and class participation (10%). The homeworks will require answering questions and programming in Matlab.
Final project (30%) The final project provides the opportunity to more deeply explore a topic of interest, individually

?

X

232x8 cell

	1	2	3	4	5	6	7
27	'COSC-1...	10	1284	'12'	12	12	7
28	'COSC-1...	10	1510	'16'	14	12	7
29	'COSC-1...	10	5057	'20'	58	136	10
30	'COSC-1...	10	1350	'16'	17	8	7
31	'COSC-2...	22	1111	'20'	4	20	3
			1474	'16'	6	36	3
			1424	'20'	7	16	5
34	'COSC-3...	30	877	'16'	4	0	5
35	'COSC-3...	31	1222	'27'	7	0	0
36	'COSC-3...	31	2105	'16'	23	4	5
37	'COSC-5...	50	2040	'20'	17	76	6
38	'COSC-5...	50	2103	'16'	17	64	12
39	'COSC-5...	55	827	'16'	1	0	0
40	'COSC-5...	58	790	'16'	6	8	1
41	'COSC-6...	60	1925	'27'	14	76	7
42	'COSC-6...	60	1160	'20'	12	40	4

FEATURE EXTRACTION

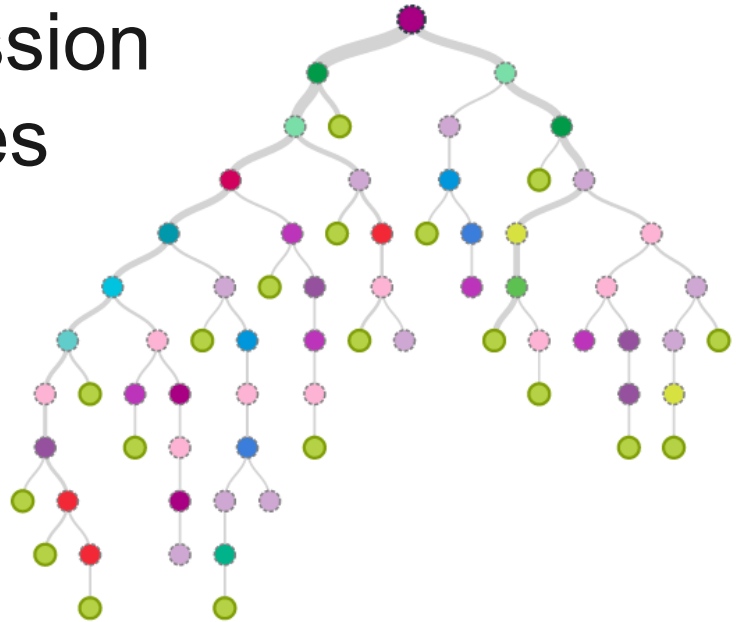
```
%% Program Description
% Program takes in formatted data and syllabus .txt files and out put
% matrices with features specified to be used with ML algorithims.
% INPUT:
% Formatted registrar office median and course information and syllabus .txt files
%
% OUTPUT:
%  X: a matrix [m x n], with each row containing a different course's features and
%  each column a different feature:
%
%     X(fileNumber,feature) := feature for given syllabus file
%     X(:,1) := Course Name
%     X(:,2) := Course Number
%     X(:,3) := Total Number Of Words In Course Syllabus
%     X(:,4) := Course Enrollment
%     X(:,5) := Number Of Negative Words
%     X(:,6) := Number of time mentioning specific words of interest such as lab,
%               homework,etc.
%     X(:,7) := Percent sign frequency
%
%  Y: a matrix [m x 1], with each row containing a different course median
%  grade:
%     Y(:,1) = courseMedian

addpath('../PDFTextExtractionCode')
regristrarCourseData = '../PDFTextExtractionCode/TEST/MedianGrades.csv';
[term, classes, enrollment, medians] = getCourseData(regristrarCourseData);
```

- 450K+ words total
- Some words hold more value
- Bulk of words are not interesting
- Word frequency/absence also determines value
- Which words are valuable?

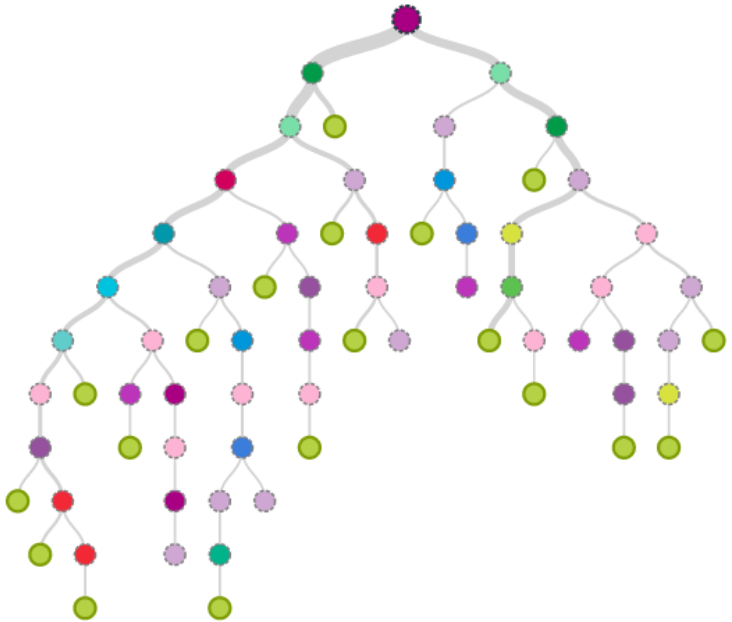
ALGORITHM

- Decision tree for regression
- Splits numerical features
- Recursive Partitioning
- C4.5 algorithm

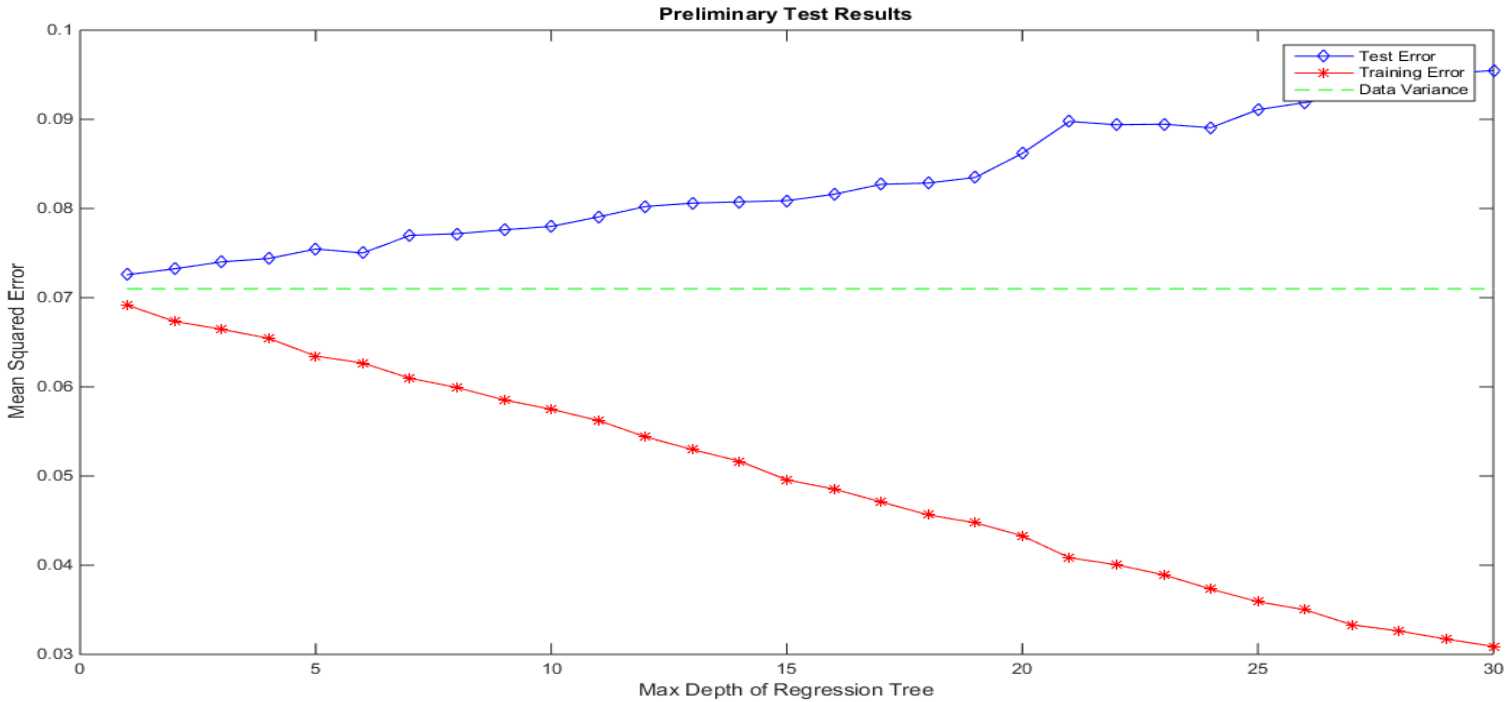


IMPLEMENTATION

```
if isStoppingCriterion()
    return regTree
else
    findBestSplit()
    regTree.insertLeftChild()
    regTree.insertRightChild()
end
```



RESULTS


Preliminary Test Results
Test Error
Training Error
Data Variance
Mean Squared Error
Max Depth of Regression Tree
0.1
0.09
0.08
0.07
0.06
0.05
0.04
0.03
0
5
10
15
20
25
30

ANY QUESTIONS?

. . . or suggestions?

Thank you for listening.