Machine Learning in Practice

Lecture 17 – Midterm Review

Plan for the Day

- Announcements
  - **Midterm!**
    Due Thursday at 11:59pm (so you get more than 24 hours)
    - After the test goes live, you are only allowed to ask questions related to technical issues in LightSide
    - Make sure you have the most recent ("beta") version: http://lightsidelabs.com/research
  - You'll download the midterm from the **Midterm 1** folder on Blackboard
  - Turn in on Blackboard
- Helpful Hints

Learn from your mistakes
on Assignments 5 and 6!

Midterm Suggestions!

- Last semester most of the points were lost on the thought questions. Refer to quiz answer keys for examples of effective answers.

- Remember proper methodology for using your development data and cross validation set
  - Careful not to look to closely at your cross validation data
  - Qualitative observations, error analysis, and feature design should be done on your **development data**
- **Don't skip the extra credit question**

LightSide

- http://lightsidelabs.com/research

Read the manual!

  - However, keep in mind that the manual can help you know how to do things you might be confused about, ***but don't use it as a methodology guide! For that refer to Carolyn's lectures as the authority!!***

- Also note that there is a Weka user manual in Part III of your textbook.

LightSide

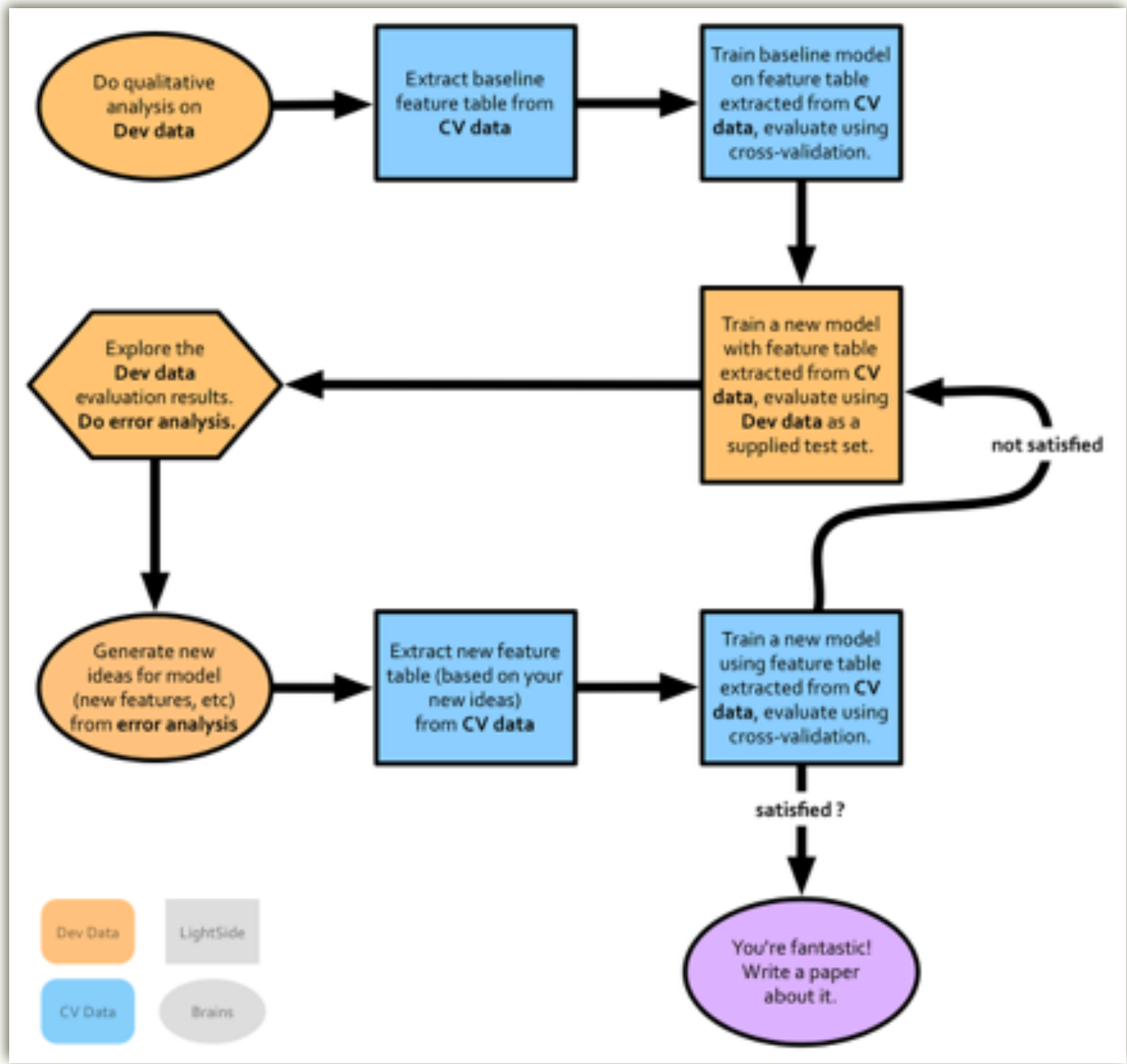
- **Basic Features changes**
  - **Count Occurrences**
    - Numeric features when checked, binary otherwise
  - “There are two ways to remove stopwords”
    - Identical for unigrams, no effect on POS bigrams.
    - **Skip Stopwords** –remove stopwords within N-grams:
      keep ***two_ways, remove_stopwords***
      but not ***there_are, are_two, ways_to, to_remove***
      (useful for content-based classification)
    - **Ignore all-stopword N-Grams** - like old “Remove Stopwords”
      Only remove N-grams made entirely of stopwords:
      keep ***are_two, are_ways, ways_to*** but not ***there_are***
      (preserves some style information)

More Helpful Hints

- On the midterm you will be asked to make qualitative observations of the data. The biggest issue is that these are typically vague – you should give specific examples and then talk about "typical" expressions you saw
  - E.g., if the task was sentiment analysis: In negative reviews **I see**… in positive reviews **I see**…

- Give specific examples of errors in your error analysis too
  - Evidence that you really did the horizontal and vertical **comparisons**
  - What were the systematic errors?
- When you say what you tried next, the ideas should be motivated by the error analysis
  - E.g., missing phrases, so I used bigrams

Iterative Model Building


Do qualitative analysis on Dev data
Extract baseline feature table from CV data
Train baseline model on feature table extracted from CV data, evaluate using cross-validation.
Train a new model with feature table extracted from CV data, evaluate using Dev data as a supplied test set.
Explore the Dev data evaluation results. Do error analysis.
not satisfied
Generate new ideas for model (new features, etc) from error analysis
Extract new feature table (based on your new ideas) from CV data
Train a new model using feature table extracted from CV data, evaluate using cross-validation.
satisfied ?
You're fantastic! Write a paper about it.
Dev Data
LightSide
CV Data
Brains

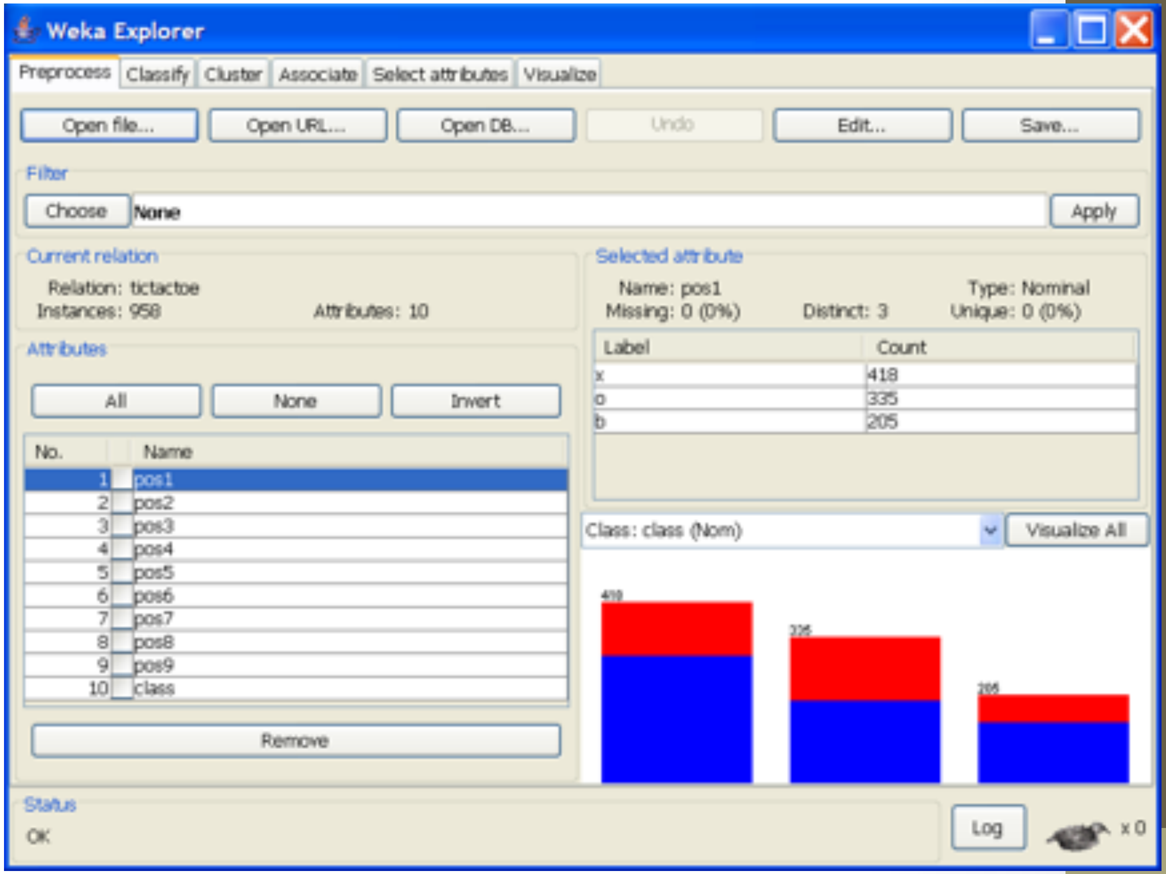
More Helpful Hints

- If you had trouble with LightSide in Assignment 6
  - Ask questions now!
  - Use a different machine
- Some people didn’t follow instructions – you need to read the instructions **carefully**!
- Need to say enough about methodology
  so Carolyn can tell if you did it right
  how you used your development and cross-validation sets,
  how you did the horizontal and vertical comparison,
  how problematic features were used in the text,
  how your ideas for improvement related back to your error analysis,
  specifics about how you tested for statistical significance.
- Also, try to **explain** your result

Tic Tac Toe:

You need to remember what we learned from this

O	X	X
X	O	O
X	O	X



Filter: None

Current relation

Relation: tictactoe
Instances: 958
Attributes: 10

Attributes

No.	Name
1	pos1
2	pos2
3	pos3
4	pos4
5	pos5
6	pos6
7	pos7
8	pos8
9	pos9
10	class

Selected attribute

Name: pos1
Type: Nominal
Missing: 0 (0%)
Distinct: 3
Unique: 0 (0%)

Label	Count
x	418
o	335
b	205

Class: class (Nom)

Status

OK

Tic Tac Toe: Remember this?

O	X	X
X	O	O
X	O	X

Decision Trees:	.67 Kappa
SMO:	.96 Kappa
Naïve Bayes:	.28 Kappa

algorithms assumption, NB not reasonable

Decision tree, overfitting, too many rules

* Remember the important message that each of these algorithms learned something different from the same data.

Feature Selection vs. Feature Design

- Feature **Selection** is an automatic (algorithmic) process to pick out the most distinguishing features.
- When evaluating via cross-validation, feature selection happens to each fold (to give you an estimate of the performance of a final feature-selected model)
- Feature **Design** is how you build your feature space, to include features that are domain-relevant or otherwise useful, possibly informed by analysis of your development data.

Feature selection and model building

Simple features
A=1, A=0, B=1, B=0

Complex features
(A=1 & B=0),
(A=1 & B=1),
…

- Feature space design, feature selection, training – all part of the model building process
- Introducing complex features allows for a simpler model to be learned
  - Nonlinear models learn contingencies between features
  - If the features themselves include the contingencies, then we can achieve the same representational power with a linear model

Feature selection and model building

- Narrowing down to a subset of features limits the space of possible models
- Many ways of distributing “labor” between these three parts of the model building process
- Principle of not training on testing data applies equally to all of these stages
  - Think about implications for user defined features
  - Don’t peek at the validation data!

questions?

Normalizing Numeric Features

- This is a (manual) process you can do to your numeric columns before building a model.
- If you want to do error analysis using the metrics we've discussed (horizontal/vertical difference, average value, model weights, etc), it's important that each feature covers the same range of values (to be comparable).
- Fit them all between 0 and 1 by subtracting the minimum and dividing by the size of the range:

- Cookie Diameter = {1, 2, 4, 5}
  = {0, 0.25, 0.75, 1.0}
- Number of Chocolate Chips = {50, 75, 100, 150}
  = {0, 0.25, 0.5, 1.0}

questions?

good luck!