HIERARCHICAL ACTIVE LEARNING (HAL) APPLICATION TO MITOCHONDRIAL DISEASE PROTEIN DATASET

James Duin

University of Nebraska–Lincoln
Master's Thesis

Spring 2017
jamesdduin@gmail.com

Introduction

- Identify the source of mutations which give rise to mitochondrial disease
- Leigh Syndrome, Leber's Hereditary Optic Neuropathy
- Proteins are hierarchically labeled according to location in the mitochondria
- Coarse-grained: learning labels near the root of the tree
- Fine-grained: learning labels towards the leaf nodes
- Learn mitochondrion concept (coarse) by combining classifiers for each target compartment (fine)

- **Active learning**: copious unlabeled data, cost associated with acquiring labels, yields best classifier for a given cost, or best classifier for minimal cost
- Previous work in text classification and and rich media indexing use hierarchies of labels to improve fine-level classification (McCallum et al. 1998, Jiang et al. 2013)
- **First, investigation of active learning in a hierarchical setting, our active over-labeling approach is shown to find the best classifier for a given budget regardless of varying label acquisition cost**

Introduction

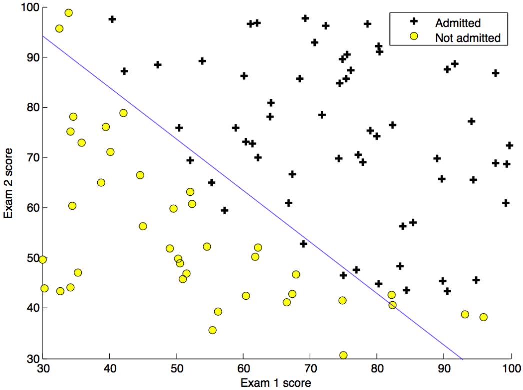
Outline:

- Machine Learning
- Active Machine Learning
- Coarse-grained vs Fine-grained Trade Off
- Active over-labeling algorithms
- Hierarchical Protein Dataset
- Application to Protein Dataset
- Experimental Results

INPUT: labeled data

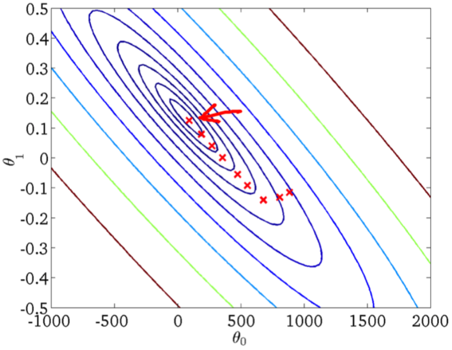
OUTPUT: learned hypothesis used to predict new instances

$h_\theta(x)$, for fixed θ_0 and θ_1 line coefficients

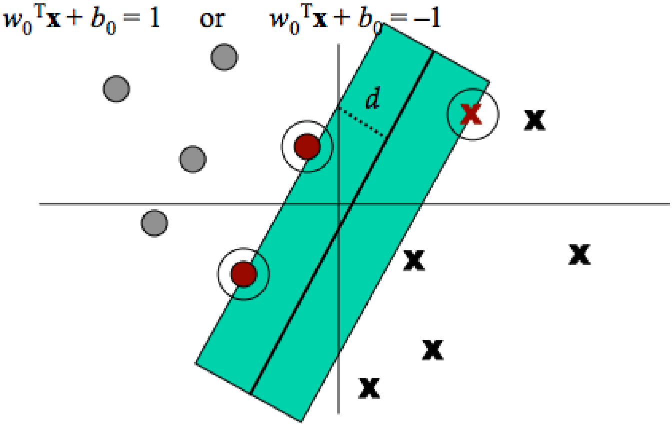


$$J(\theta_0, \theta_1)$$

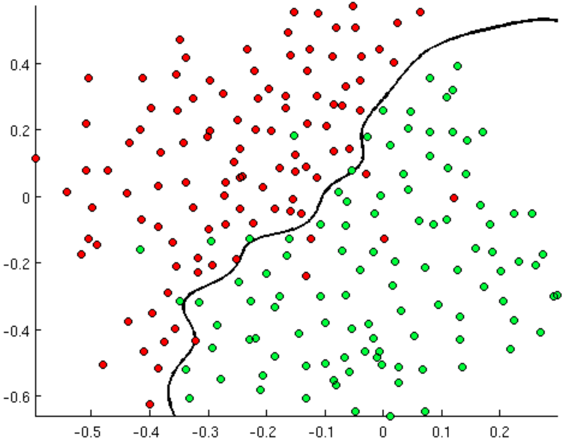
(function of the parameters θ_0, θ_1)



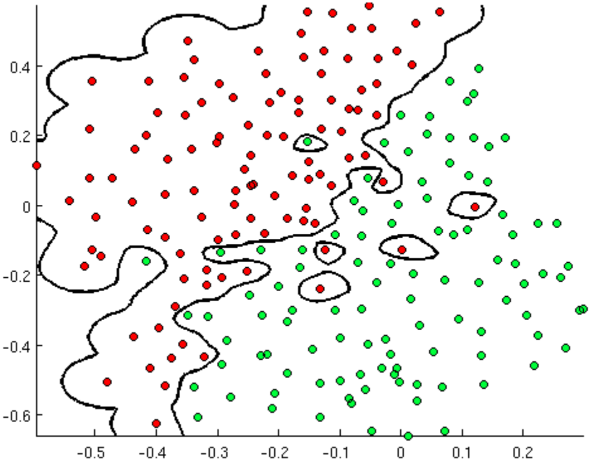
And SVM constructs a line, plane or hyperplane that separates the features with the greatest margin.



The greater the functional margin the lower the generalization error of the classifier

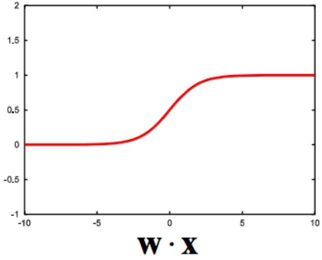


Kernel functions implicitly map inputs into high-dimensional feature spaces



Logistic Regression (Logit) estimates the probability of a binary response, learns coefficients **w** of the input vector **x** and passes dot product through sigmoid function. (Maximum likelihood learning)

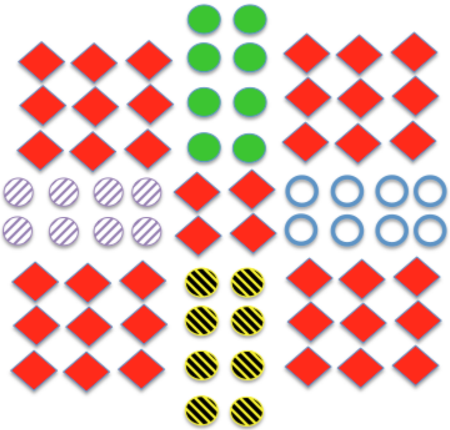
$$g(\mathbf{x}, \mathbf{w}) = \frac{1}{1 + \exp(-\mathbf{w} \cdot \mathbf{x})}$$



Active Machine Learning

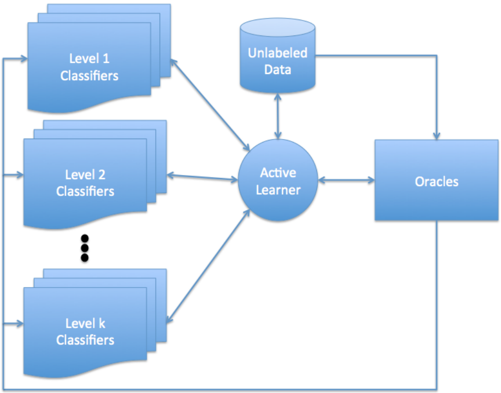
- Unlabeled data is abundant but manually labeling is expensive, e.g. text categorization, drug discovery
- The learner queries an **oracle** or **supervisor** which labels the data at a certain cost
- Active learning solicits new instances that can maximally improve performance of the learned classifier, e.g., uncertainty sampling
- Learns the best performing classifier for the minimal amount of labeling cost, or for a given purchase budget
- Acquires labels for each level of the hierachy at a certain cost, spends according to a purchase budget

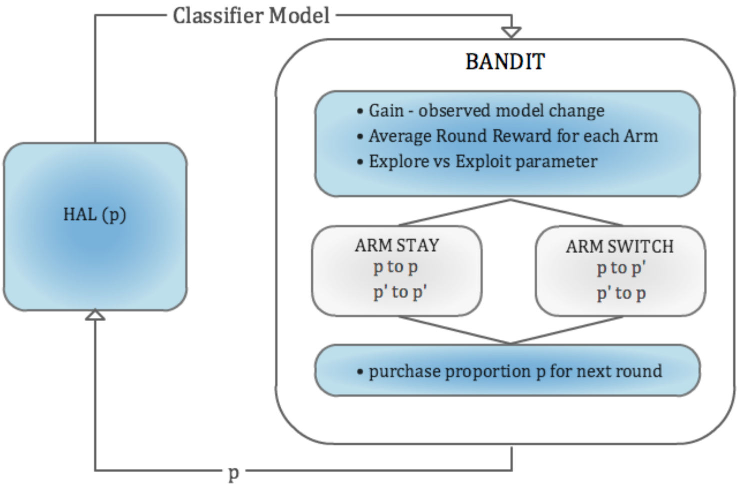
Active over-labeling solicits labels at a finer level of granularity than the target concept



Hierarchical Active Learning (HAL)

INPUT: purchase proportion p



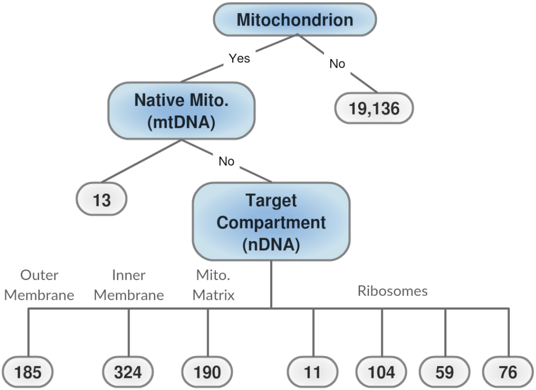

Classifier Model
BANDIT
• Gain - observed model change
• Average Round Reward for each Arm
• Explore vs Exploit parameter
HAL (p)
ARM STAY
p to p
p' to p'
ARM SWITCH
p to p'
p' to p
• purchase proportion p for next round
p

Hierarchical Bioinformatics Data Set

Feature Sources

- Mitoproteome: database of human mitochondrial proteins
- SwissProt: database of experimentally validated human proteins

Type of Properties	Features	Sources
General sequence features	Amino acid composition, sequence length, etc.	Cui et al, PROFEAT
Physico chemical properties	Hydrophobicity, polarity, etc.	Cui et al, PROSO, Phoebus
Structural properties	Secondary structural content, shape, etc.	SSCP
Domains and motifs	Signal peptide, transmembrane domains, etc.	SignalP, TMB-Hunt, NetOgly, TatP


Mitochondrion
Yes
No
Native Mito. (mtDNA)
19,136
No
13
Target Compartment (nDNA)
Outer Membrane
Inner Membrane
Mito. Matrix
Ribosomes
185
324
190
11
104
59
76

Training and Testing Coarse-Grain and Fine-Grain Classifiers

Number of proteins in each class:

Classes	Count	Totals
Non Mito 0	19136	All: 20098
mtDNA 1	13	Coarse: 19136
nDNA 2	185	Fine: 962
nDNA 3	324	Features: 449
nDNA 4	190	
nDNA 5	11	
nDNA 6	104	
nDNA 7	59	
nDNA 8	76	

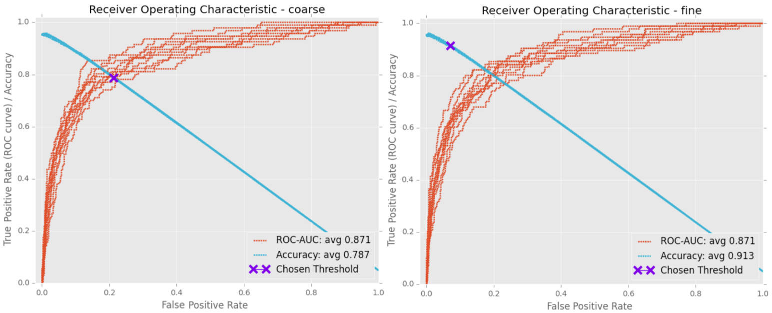
SVM and Logit Classifier Performance

Conventional ML

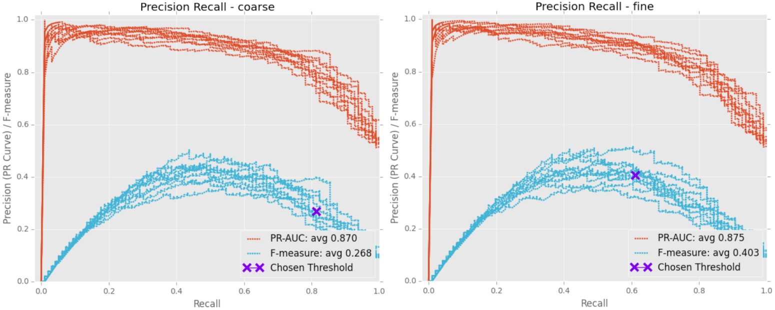
- Tuned parameters for SVM and Logit via an independent run of cross-validation
- **Accuracy**: the percentage of correctly classified results
- **Precision**: a measure of result relevancy
- **Recall**: a measure of how many truly relevant results are returned
- **F-measure**: the harmonic mean of precision and recall
- **PR curve**: plot precision and recall as classifier threshold is varied
- **ROC curve**: plot false positive rate and true positive rate as classifier threshold is varied
- **AUC**: area under the curve, both curves have an optimal AUC of 1.0

SVM and Logit Classifier Performance

Accuracy Analysis (Logit)

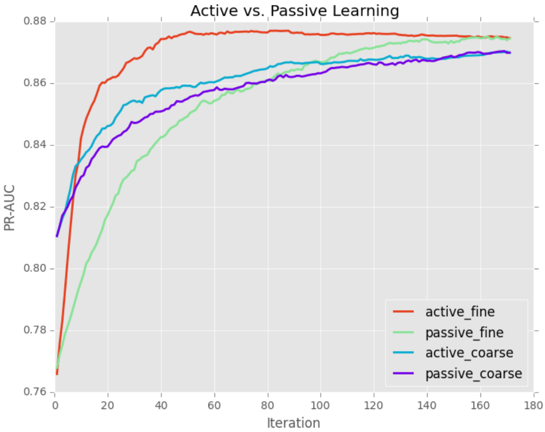


SVM and Logit Classifier Performance
F-measure Analysis (Logit)



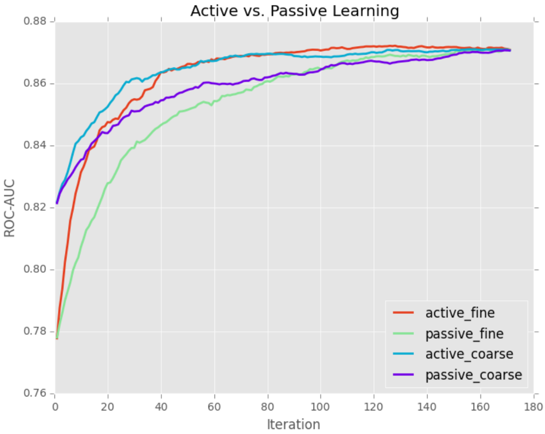
Active vs. Passive Curve Analysis
Logit PR-AUC curves

Iteration: a cycle of the HAL algorithm, a single round of purchasing labels

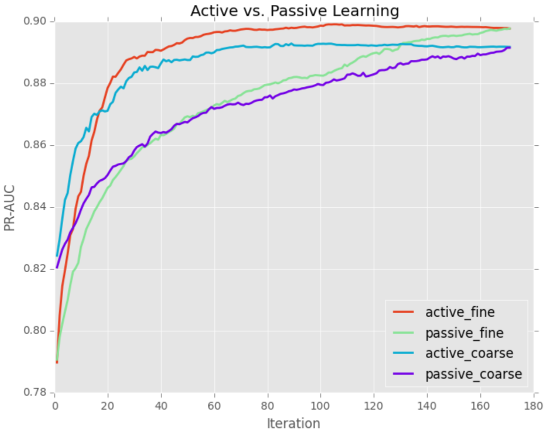


Active vs. Passive Curve Analysis

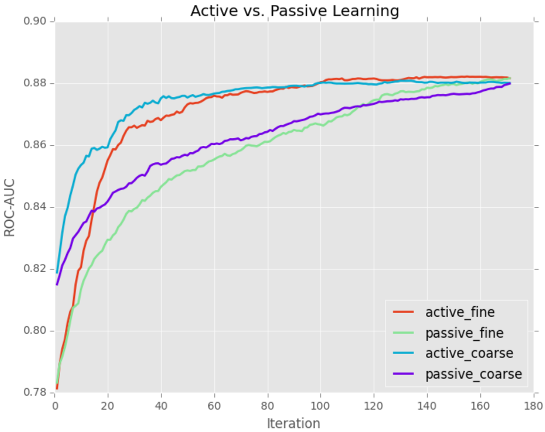
Logit ROC-AUC curves



Active vs. Passive Curve Analysis
SVM PR-AUC curves

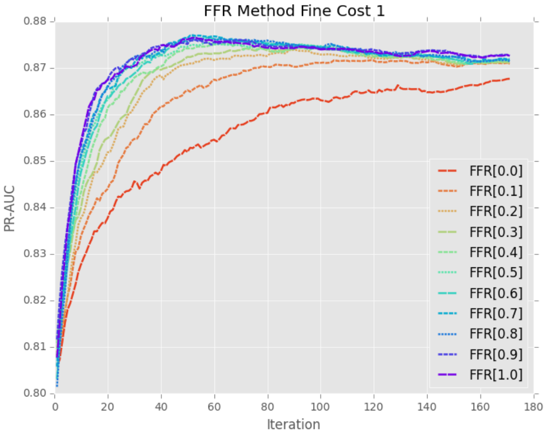


Active vs. Passive Curve Analysis
SVM ROC-AUC curves



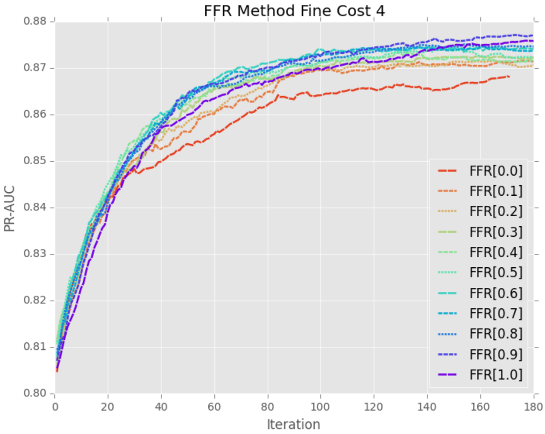
Plots for Fine Fixed Ratio Results

Successive iterations of HAL with fine cost of 1 and coarse cost of 1



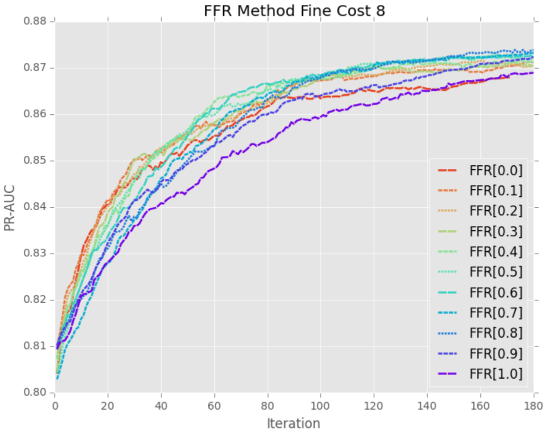
Plots for Fine Fixed Ratio Results

Successive iterations of HAL with fine cost of 4



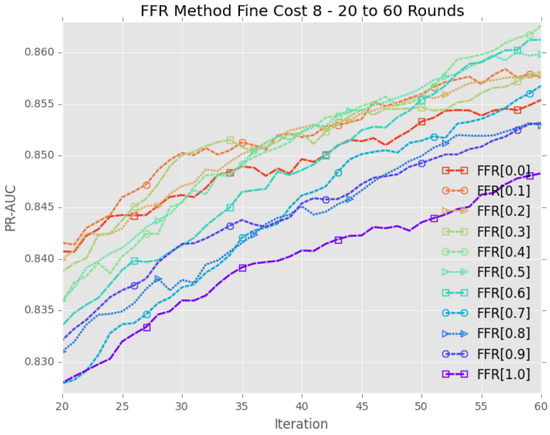
Plots for Fine Fixed Ratio Results

Successive iterations of HAL with fine cost of 8



Plots for Fine Fixed Ratio Results

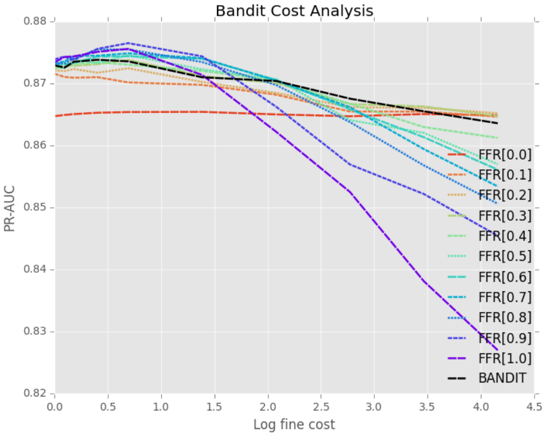
Expanded view Fine Cost 8 - Rnds 20 to 60



BANDIT Approach Results

Varying Cost Analysis

- The BANDIT approach is compared to the previous FFR curves for the following fine-grain costs $\{1.0, 1.1, 1.2, 1.5, 2.0, 4.0, 8.0, 16.0, 32.0, 64.0\}$
- Budget held fixed at round 120.
- The metric *diff* is the learner's absolute difference in PR-AUC from the top learner for a given cost.
- The metric *rank* is the learners 0 indexed ranking in terms of PR-AUC for a given cost.


Bandit Cost Analysis
PR-AUC
0.88
0.87
0.86
0.85
0.84
0.83
0.82
0.0
0.5
1.0
1.5
2.0
2.5
3.0
3.5
4.0
4.5
Log fine cost
FFR[0.0]
FFR[0.1]
FFR[0.2]
FFR[0.3]
FFR[0.4]
FFR[0.5]
FFR[0.6]
FFR[0.7]
FFR[0.8]
FFR[0.9]
FFR[1.0]
BANDIT

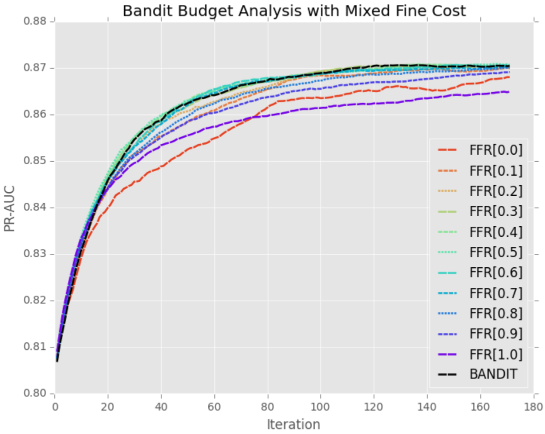
BANDIT Approach Results

Varying Cost Analysis - Rank and Diff Metrics

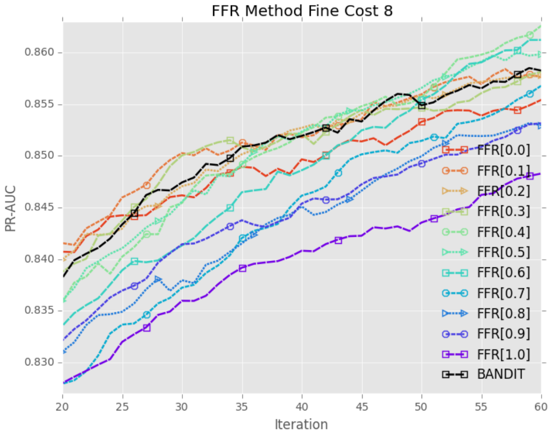
	diff				rank			
	min	max	mean	std	min	max	mean	std
algorithm								
BANDIT	0.000	0.003	**0.001**	0.001	0	8	4.8	2.315
FFR[0.0]	0.000	0.011	0.007	0.004	1	11	8.8	3.429
FFR[0.1]	0.001	0.006	0.003	0.002	3	10	8.0	2.793
FFR[0.2]	0.000	0.004	0.002	0.001	0	9	6.5	3.500
FFR[0.3]	0.000	0.003	0.001	0.001	0	8	5.1	2.663
FFR[0.4]	0.000	0.004	0.002	0.001	1	8	5.6	2.200
FFR[0.5]	0.000	0.008	0.002	0.002	0	8	4.6	2.200
FFR[0.6]	0.000	0.009	0.002	0.003	1	7	4.6	1.855
FFR[0.7]	0.000	0.012	0.002	0.004	0	8	**3.3**	2.571
FFR[0.8]	0.000	0.015	0.003	0.005	1	9	4.8	3.027
FFR[0.9]	0.000	0.020	0.005	0.007	0	10	4.3	4.605
FFR[1.0]	0.000	0.038	0.009	0.013	1	11	5.6	4.630

BANDIT Approach Results

Varying Budget Analysis - Mixed Cost



BANDIT Approach Results
BANDIT - Rnds 20 to 60



Related Work

Results agree with Mo et al.'s experiments:

- Synthetic Dataset with Gradient Boosted Regression Trees
- Reuters Corpus Volume Text Categorization Dataset with Logit
- Richmond Daily Dispatch Sequence Tagging Dataset with Conditional Random Fields

Conclusions

- Fine outperforms Coarse in PR-AUC
- Active outperforms Passive in PR-AUC
- HAL ran with variable cost, fine proportions and budget
- BANDIT approach shown to be robust to changes in cost and budget

Future Work

- Future work is to apply the active over-labeling approach to other datasets with more complex hierarchical label trees; datasets derived from Gene Ontology research could be investigated

Acknowledgements

- I would like to thank my advisor Dr. Stephen Scott, Yugi Mo and Dr. Douglas Downey for continued guidance. I would like to thank Dr. Juan Cui and Dr. Ashok Samal for serving on my comittee. Additionally, I would like to thank Jiang Shu and Kevin Chiang for their assistance accessing and understanding the protein dataset.

Questions

jamesdduin@gmail.com

Plots for Fine Fixed Ratio Results

Successive iterations of HAL with fine cost of 16

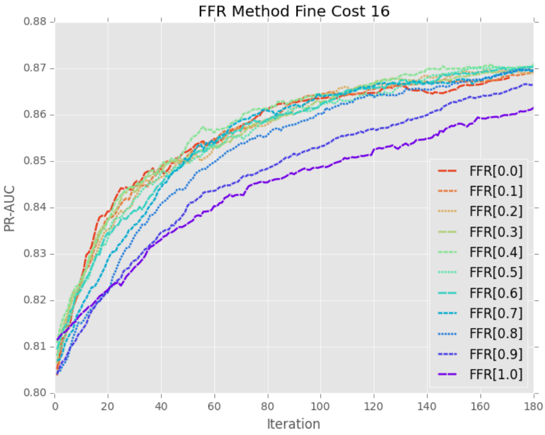
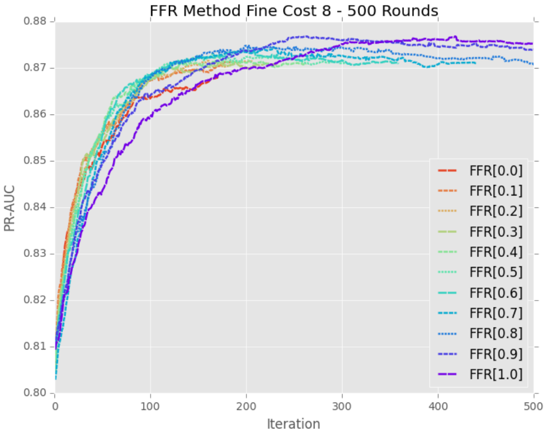


Figure: The fine cost is increased to 16. The fine cost is to high to offset the decreased number of instances purchased.

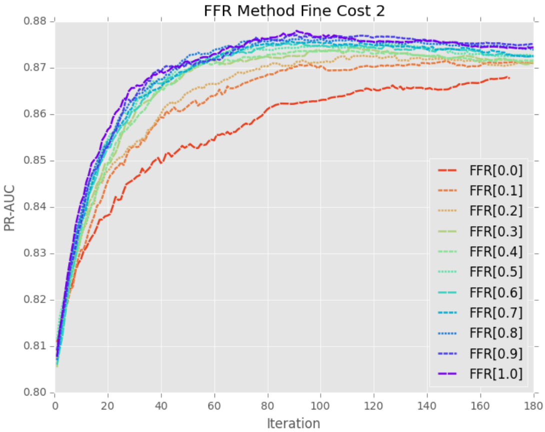
Plots for Fine Fixed Ratio Results

Fine Cost 8 - Rnds to 500



Plots for Fine Fixed Ratio Results

Successive iterations of HAL with fine cost of 2



Evaluating Classifier Performance
Confusion Matrix

Divide data into train and a test set. Analyze test set with the following values:

- True-Negatives (T_n): Correctly classified negatives
- False-Negatives (F_p): Incorrectly classified negatives
- False-Positives (F_n): Incorrectly classified positives
- True-Positives (T_p): Correctly classified positives

Example of a confusion matrix for a test set with 100 negatives and 50 positives:

conf (T_n/F_n)	conf (F_p/T_p)
90	10
20	30

Precision is a measure of result relevancy:

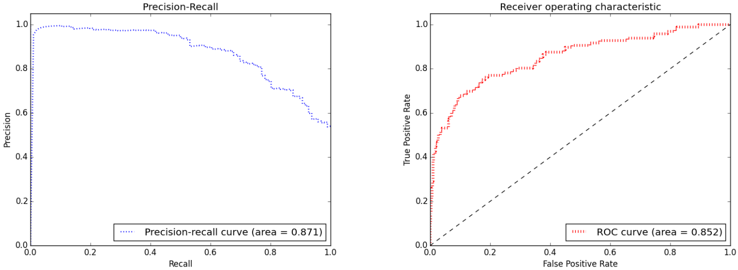
$$P = \frac{T_p}{T_p + F_p} \tag{1}$$

Recall is a measure of how many truly relevant results are returned:

$$R = \frac{T_p}{T_p + F_n} \tag{2}$$

The F-measure or F1-measure (F1) is the harmonic mean of precision and recall:

$$F1 = 2 \cdot \frac{P \cdot R}{P + R} \tag{3}$$



(a) PR curve.

(b) ROC curve.

Figure: Examples of PR and ROC curves with their corresponding AUC values.

Training and Testing Coarse-Grain and Fine-Grain Classifiers

Table: Number of proteins in each partition:

Folds	All	0	1	2	3	4	5	6	7	8
1	2010	1914	1	19	32	19	1	11	6	7
2	2010	1914	1	19	32	19	1	11	6	7
3	2010	1914	1	19	32	19	1	11	5	8
4	2010	1914	1	19	32	19	1	10	6	8
5	2010	1914	1	18	33	19	1	10	6	8
6	2010	1914	1	18	33	19	1	10	6	8
7	2010	1913	2	18	33	19	1	10	6	8
8	2010	1913	2	18	33	19	1	10	6	8
9	2009	1913	2	18	32	19	2	10	6	7
10	2009	1913	1	19	32	19	1	11	6	7
Total	20098	19136	13	185	324	190	11	104	59	76

Training and Testing Coarse-Grain and Fine-Grain Classifiers

The following variables were varied for both SVM and Logit classifiers:

- Preprocessing Scaling Methods
- Preprocessing Feature Selection
- Class Weight
- SVM Kernel, Cost, and Gamma parameters
- Logit Cost, Fine class weights, Tolerance

SVM and Logit Classifier Performance

Conventional ML

Table: Logit results after parameter tuning:

Title	PR	ROC	Acc	F1	conf (tn/fn)	conf (fp/tp)
coarse	0.870	0.871	0.787	0.268	(1503.2 / 17.8)	(410.4 / 78.3)
fine	0.875	0.871	0.913	0.403	(1776.5 / 37.3)	(137.1 / 58.8)

Table: SVM results after parameter tuning:

Title	PR	ROC	Acc	F1	conf (tn/fn)	conf (fp/tp)
coarse	0.892	0.880	0.866	0.347	(1669.5 / 24.8)	(244.1 / 71.3)
fine	0.898	0.882	0.942	0.485	(1839.0 / 41.5)	(74.6 / 54.6)

Active vs. Passive Curve Analysis
Logit Accuracy

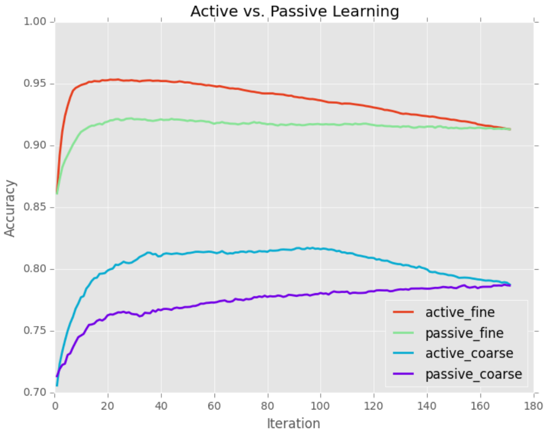


Figure: The accuracy of the classifiers stays at roughly the same rate throughout the rounds; this is due to an effective weighting scheme.

Active vs. Passive Curve Analysis

Logit F-measure

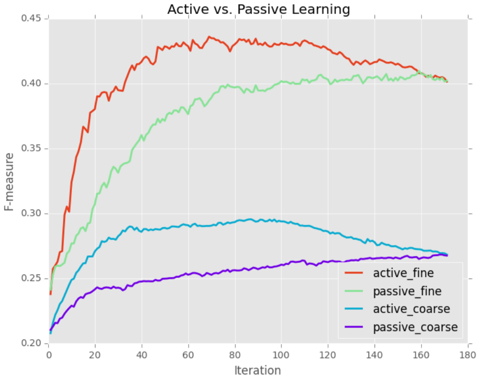


Figure: Both curves show a dominance of fine over coarse and Active over Passive.

Dynamically Adapting Purchase Proportions p or p'

- For round n, calculate gain g in terms of observed model change
- Calculate average round reward for each arm
- Calculate $\varepsilon_n = \min\left\{1, \frac{2}{n}\right\}$
- With probability $1 - \varepsilon_n$ play arm with highest current average reward for round n, otherwise explore
- After playing arm, run HAL with chosen p or p'

ARM STAY	ARM SWITCH
$r(n) = 0$	$r(n) = \begin{cases} -g(n)/\lvert g(n)\rvert & \text{if } p \to p' \\ g(n)/\lvert g(n)\rvert & \text{if } p' \to p \\ 0 & \text{if } p \to p \text{ or } p' \to p' \end{cases}$

Bibliography

- Y. Mo, S. D. Scott, and D. Downey, Learning hierarchically decomposable concepts with active over-labeling, in 2016 IEEE 16th International Conference on Data Mining (ICDM), Dec 2016, pp. 340349.
- J. Z. Juan Cui, Kevin Chiang, Prediction of nuclear and locally encoded mitochondrion. Lincoln, NE: Nebraska Gateway to Nutrigenomics 6th Annual Retreat, June 9 2014. [Online]. Available: http://cehs.unl.edu/nutrigenomics/nebraska-gateway-nutrigenomics-6th-annual-retreat/
- T. M. Mitchell, Machine Learning, 1st ed. New York, NY, USA: McGraw-Hill, Inc., 1997.

Bibliography

- L. Buitinck, G. Louppe, M. Blondel, F. Pedregosa, A. Mueller, O. Grisel, V. Niculae, P. Prettenhofer, A. Gramfort, J. Grobler, R. Layton, J. VanderPlas, A. Joly, B. Holt, and G. Varoquaux, API design for machine learning software: experiences from the scikit-learn project, in ECML PKDD Workshop: Languages for Data Mining and Machine Learning, 2013, pp. 108122
- D. Cotter, P. Guda, E. Fahy, and S. Subramaniam, Mitoproteome: mitochondrial protein sequence database and annotation system, Nucleic Acids Research, vol. 32, no. suppl1, p. D463, 2004. [Online]. Available: +http://dx.doi.org/10.1093/nar/gkh048

Bibliography

- J. Cui, L. Y. Han, H. Li, C. Y. Ung, Z. Q. Tang, C. J. Zheng, Z. W. Cao, and Y. Z. Chen, Computer prediction of allergen proteins from sequence-derived protein structural and physicochemical properties, Molecular Immunol-
ogy, vol. 44, no. 4, pp. 514 520, 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0161589
- A. McCallum, R. Rosenfeld, T. M. Mitchell, and A. Y. Ng, Improving text classification by shrinkage in a hierarchy of classes, in Proceedings of the Fifteenth International Conference on Machine Learning, ser. ICML 98. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 1998, pp. 359367. [Online]. Available: http://dl.acm.org/citation.cfm?id=645527.657461

Bibliography

- W. Jiang and Z. W. Ras, Multi-label automatic indexing of music by cascade classifiers, Web Intelli. and Agent Sys., vol. 11, no. 2, pp. 149170, Apr. 2013. [Online]. Available: http://dl.acm.org/citation.cfm?id=2590084.2590088
- etc.