Nebraska Lincoln

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

- Machine Learning
- Evaluating Classifier Performance
- Hierarchical Bioinformatics Dataset
- Coarse-grained vs Fine-grained Trade Off
- Active Over-Labeling
- Hierarchical Active Learning
- Dynamically Adapting Purchase Proportions
- Related Work
- Training and Testing Coarse-grained and Fine-grained Classifiers
- SVM and Logit Classifier Performance
- Active vs Passive Curve Analysis
- Plots for Fine Fixed Ratio Results
- BANDIT Approach Results
- Conclusions and Future Work

Machine Learning

- Machine learning (ML) algorithms are defined as computer programs that learn from experience E with respect to some class of tasks T and performance measure P, if their performance at tasks in T, as measured by P, improves with experience E - *Mitchell*.
- Support Vector Machine
- Logistic Regression

Evaluating Classifier Performance

Confusion Matrix

- True-Negatives (T_n): Correctly classified negative instances.
- False-Negatives (F_p): Incorrectly classified negative instances.
- False-Positives (F_n): Incorrectly classified positive instances.
- True-Positives (T_p): Correctly classified positive instances.

Table: Example of a confusion matrix, with 100 negative and 50 positive instances in the test set.

conf (tn/fn)	conf (fp/tp)
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:

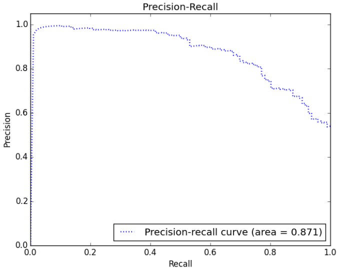
$$R = \frac{T_p}{T_p + F_n} \tag{2}$$

Evaluating Classifier Performance

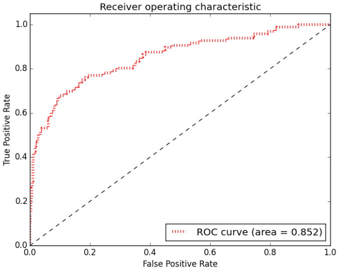
F-Measure

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.

Table: Features of the protein dataset along with their respective sources.

Type of Properties	Features	Sources
General sequence features	Amino acid composition, sequence length, etc.	Calculated by Kevin Chiang at UNL
Physico chemical properties	Hydrophobicity, polarity, etc.	Computed from Cui et al.
Structural properties	Secondary structural content, shape, etc.	SSCP
Domains and motifs	Signal peptide, transmembrane domains, etc.	SignalP, NetOgly

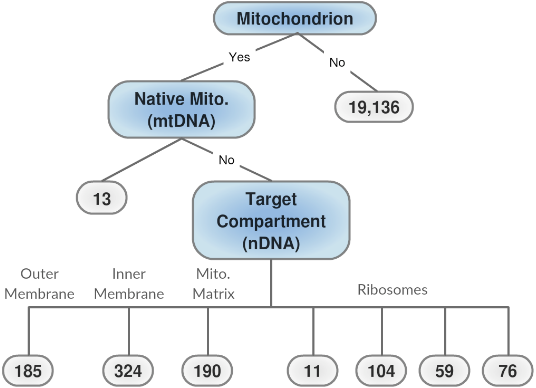


Figure: The protein dataset hierarchy of labels along with the instance count for each label.

Coarse-grained vs Fine-grained Trade Off

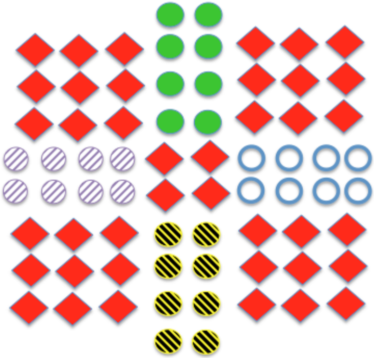


Figure: Demonstration of a dataset that would benefit from multiple fine-grained learners for each circle type, from Mo et al.

Active Over-Labeling

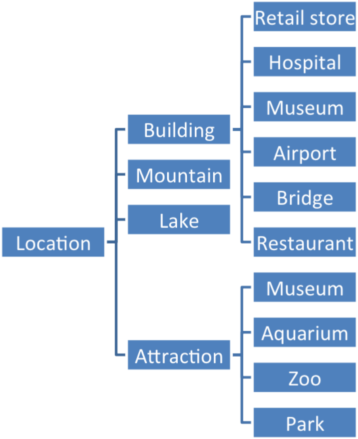


Figure: A labeling tree based on the text categorization dataset RCV1, from Mo et al.

Hierarchical Active Learning

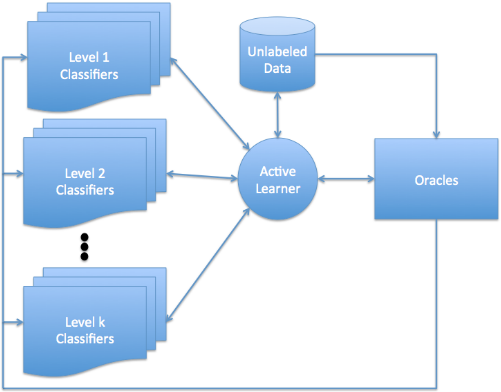


Figure: Diagram of HAL approach

Dynamically Adapting Purchase Proportions

- HAL is a fixed-fine ratio methodology.
- It takes as input a purchase proportion vector p, which specifies how much of the budget should be used to purchase at a given level in the hierarchy.
- The task of choosing the level of granularity to purchase labels is framed as a multi-armed bandit problem, and solved using Auer et al.'s ϵ-greedy bandit algorithm (BANDIT) From Auer et al.

Related Work

- The experiments and methods described in this work demonstrate how leveraging fine-grained label information can improve the accuracy of a coarse-grained (root-level) classifier, and investigate active learning in a hierarchical setting where label acquisition cost can vary, from Mo et al.

Application to Dispatch Dataset

Analysis and evaluation follow Mo et al.'s work.

- 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

Training and Testing Coarse-Grain and Fine-Grain Classifiers

Table: Class Totals

Classes	Count
0	19136
1	13
2	185
3	324
4	190
5	11
6	104
7	59
8	76
Tot All	20098
Tot Coarse	19136
Tot Fine	962
Features	449

Training and Testing Coarse-Grain and Fine-Grain Classifiers

Table: Example Fold Totals

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 entire dataset results after parameter tuning

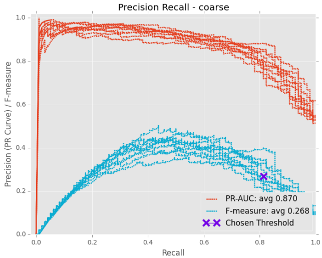
Title	PR	ROC	Acc	F1	conf (tn/fn)	conf (fp/tp)
coarse	0.870	0.871	0.787	0.268	(1503.2 / 17.8)	(410.4 / 78.3)
fine	0.875	0.871	0.913	0.403	(1776.5 / 37.3)	(137.1 / 58.8)

Table: SVM entire dataset results after parameter tuning

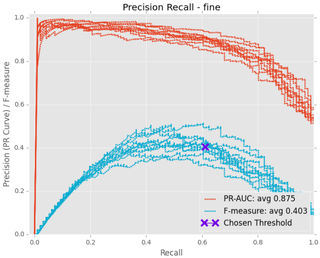
Title	PR	ROC	Acc	F1	conf (tn/fn)	conf (fp/tp)
coarse	0.892	0.880	0.866	0.347	(1669.5 / 24.8)	(244.1 / 71.3)
fine	0.898	0.882	0.942	0.485	(1839.0 / 41.5)	(74.6 / 54.6)

SVM and Logit Classifier Performance

F-measure Analysis



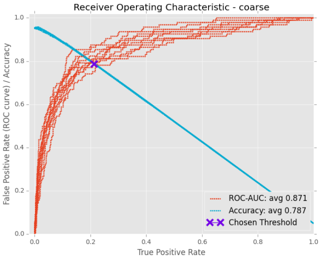
(a) Log Reg Pr Curves - Coarse



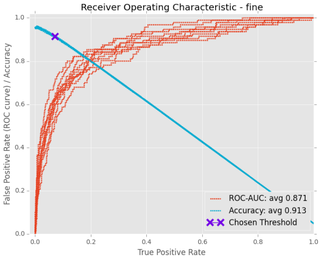
(b) Log Reg Pr Curves - Fine

Figure: The fine default threshold occurs at a point on the PR curve associated with a higher F-measure score compared to the coarse curves.

SVM and Logit Classifier Performance
F-measure Analysis



(a) Log Reg ROC Curves - coarse



(b) Log Reg ROC Curves - fine

Figure: Fine has a higher accuracy than coarse at the default threshold for the Logit classifier.

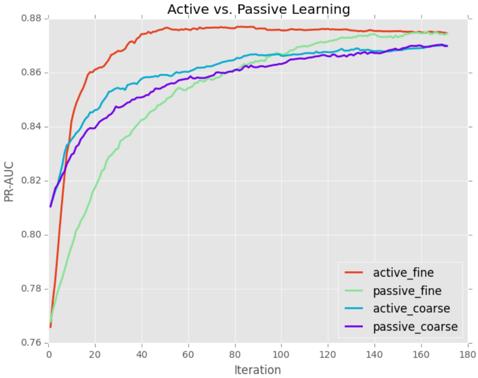


Figure: The PR-AUC curves for rounds with the Logistic Regression classifier conforms to expectations, with active fine having the best performance, and Active outperforming Passive for both coarse and fine classifier types.

Active vs. Passive Curve Analysis

Logit ROC-AUC curves

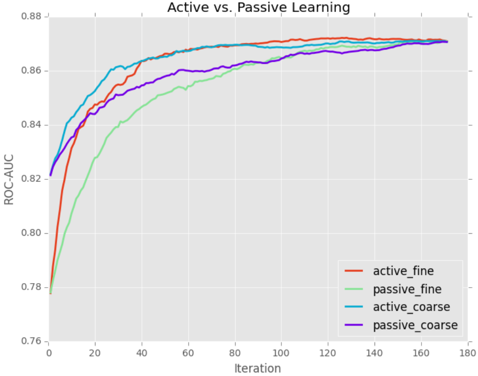


Figure: The ROC-AUC curves for rounds with the Logistic Regression classifier. The active curves beat out the passive curves for both coarse and fine.

Active vs. Passive Curve Analysis

Logit Accuracy

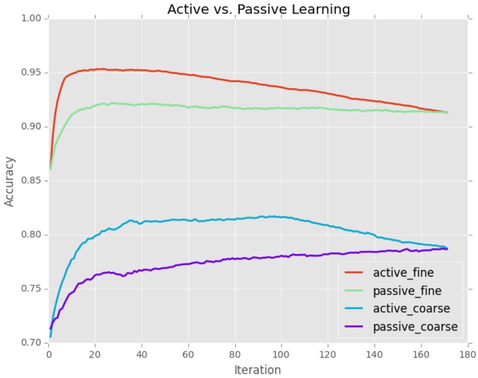


Figure: The accuracy of the classifiers stays at roughly the same rate throughout the rounds; this is due to an effective weighting scheme.

Active vs. Passive Curve Analysis
Logit F-measure

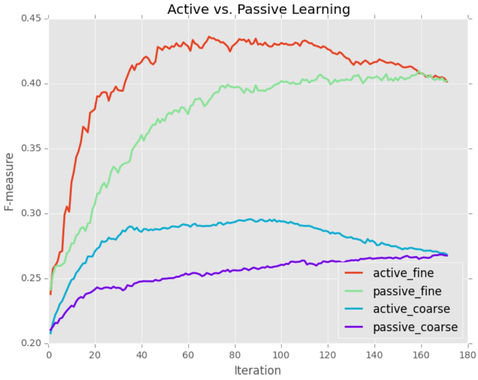


Figure: Both curves show a dominance of fine over coarse and Active over Passive.

Active vs. Passive Curve Analysis
SVM PR-AUC curves

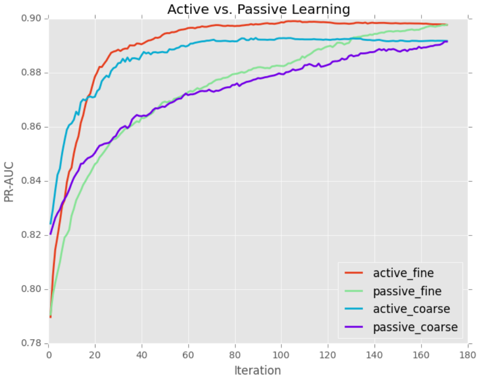


Figure: The PR AUC curves for SVM show a slight advantage for active fine, similar to the Logit results.

Active vs. Passive Curve Analysis
SVM ROC-AUC curves

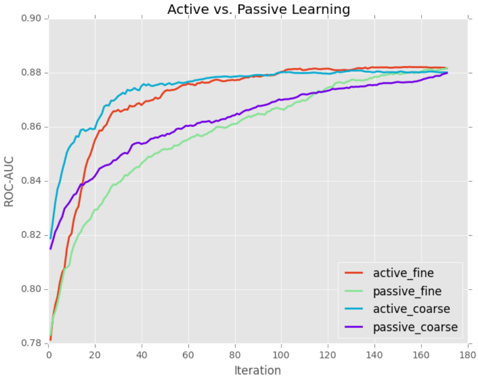


Figure: The ROC AUC curves for SVM match the Logit results, the convergence of active fine to active coarse takes slightly longer, round 60 compared to round 40.

Plots for Fine Fixed Ratio Results

Fine Cost 1

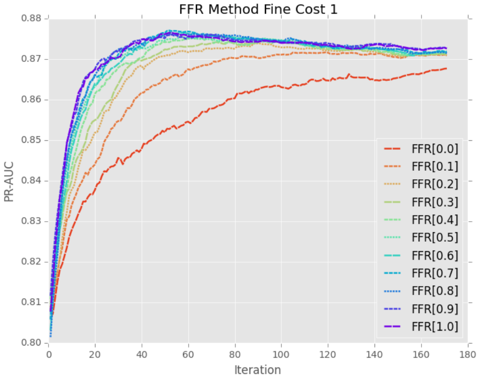


Figure: For this curve the fine and coarse grain labels both have a cost of 1.

Plots for Fine Fixed Ratio Results
Fine Cost 2

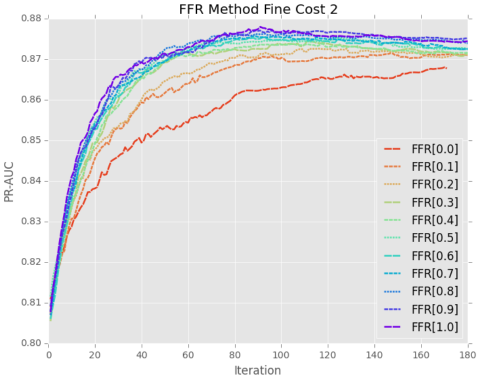


Figure: At fine cost 2, advantage of the higher FFR values decreases but the ordering of the curves remains unchanged.

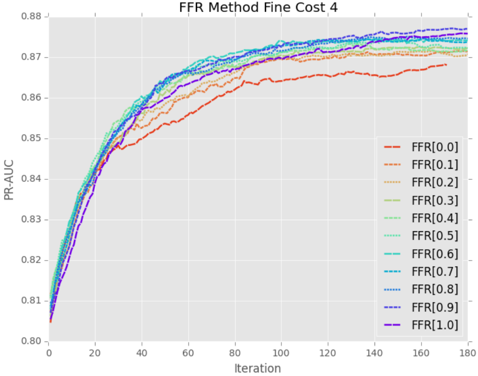


Figure: At fine cost 4, the highest FFR 1.0 is no longer preferred, the cost is to high for fine instances PR-AUC utility to overcome the PR-AUC increase gained by purchasing more coarse instances.

Plots for Fine Fixed Ratio Results

Fine Cost 8

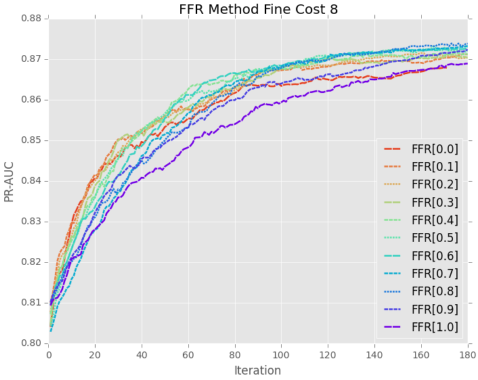


Figure: At fine cost 8 the middle FFR values outperform the extreme values for rounds 0 to 180.

Plots for Fine Fixed Ratio Results

Fine Cost 8 - Rnds to 500

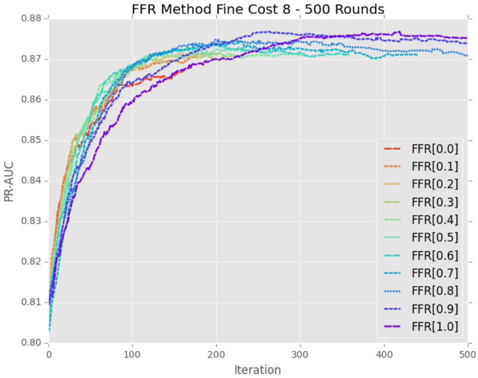


Figure: This shows the iterations continuing through round 500, the curves with the higher fine rates eventually settle to the same end point that the curves with the high rates of coarse labels purchased achieved at previous iterations.

Plots for Fine Fixed Ratio Results

Fine Cost 8 - Rnds 20 to 60

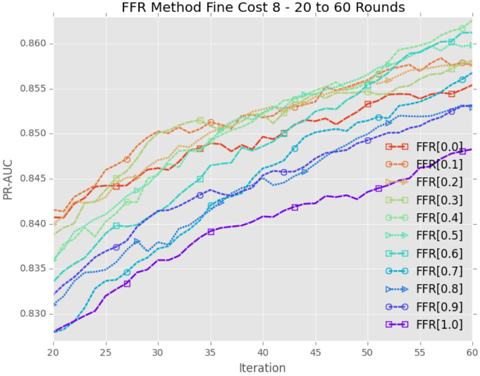


Figure: The fine cost 8 curves shown expanding the rounds 20-60. If a round budget of 40 occurs than the recommended FFR would be 0.2.

Plots for Fine Fixed Ratio Results

Fine Cost 16

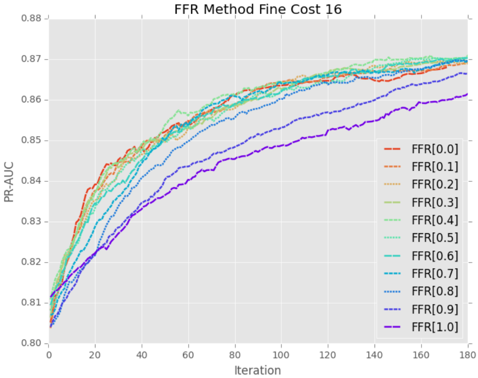


Figure: The fine cost is increased to 16. The cost is to high for the fine label advantage to offset the decreased number of instances purchased.

BANDIT Approach Results

Varying Cost Analysis - Plot

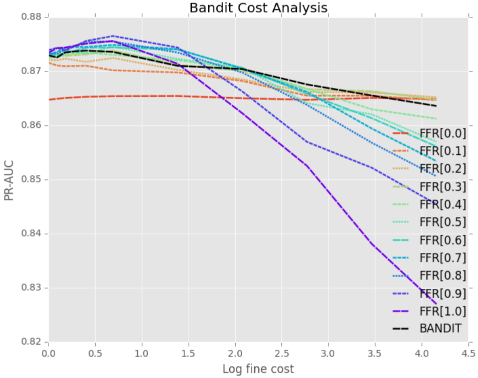


Figure: BANDIT log fine cost analysis with budget fixed.

BANDIT Approach Results

Varying Cost Analysis - Rank and Diff Metrics

Table: Aggregated PR AUC for the protein dataset

	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

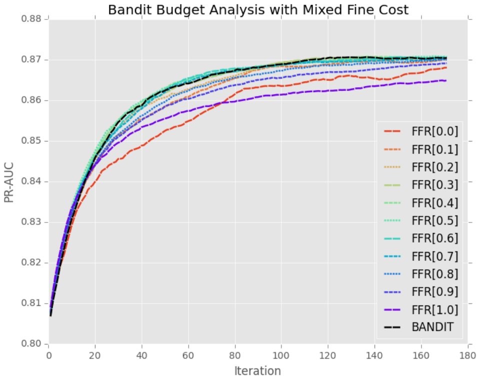


Figure: BANDIT mixed fine cost plot.

BANDIT Approach Results
BANDIT - Rnds 20 to 60

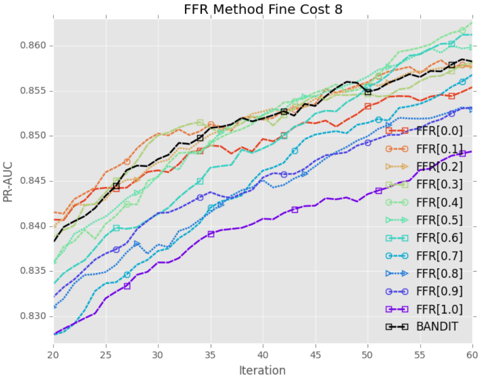


Figure: The fine cost 8 curves shown expanding the rounds 20-60. With the BANDIT approach plotted. At budget iteration 40, BANDIT PR-AUC is within 0.0007 of the top learner's PR-AUC.

Conclusions and 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

Questions

jamesdduin@gmail.com

Nebraska Lincoln

Bibliography

- 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.
- 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.

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/S01615890(
- etc.