Differential Prediction Using Inductive Logic Programming

Houssam Nassif

Thesis Proposal
14 January 2011

Outline

1. Motivation
   - Differential Prediction (DP)
   - Inductive Logic Programming (ILP)
   - Applications
2. Preliminary Results
   - Predicting Hexose Binding Sites
   - DP for Invasive/In-Situ
   - BI-RADS Information Extraction
3. Proposed Work
   - Differential Predictive Rules Definition
   - DP within the ILP Framework
   - Randomizing Recall
   - BI-RADS Terms Annotation
4. Wrap-Up

Outline

1. Motivation
   - Differential Prediction (DP)
   - Inductive Logic Programming (ILP)
   - Applications
2. Preliminary Results
   - Predicting Hexose Binding Sites
   - DP for Invasive/In-Situ
   - BI-RADS Information Extraction
3. Proposed Work
   - Differential Predictive Rules Definition
   - DP within the ILP Framework
   - Randomizing Recall
   - BI-RADS Terms Annotation
4. Wrap-Up

Breast-Cancer Stages

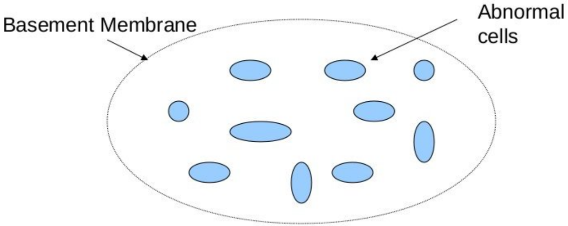


Figure: In-Situ Cancer Stage

Breast-Cancer Stages

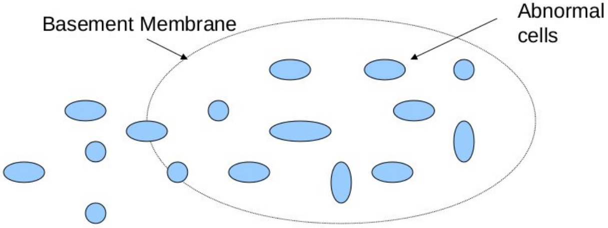


Figure: Invasive Cancer Stage

Cancer Stage Features

- In Situ can develop into Invasive
  - Current practice: Always treat In Situ
- Time to spread may be very long
  - Over-diagnosis (unnecessary treatment)
  - Patient may die of other causes

- What features characterize In Situ in older patients?
- What features change between older and younger?

Cancer Stage Features

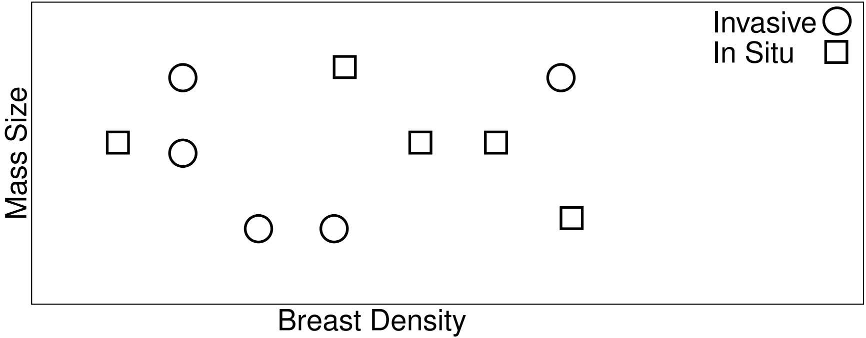
- In Situ can develop into Invasive
  - Current practice: Always treat In Situ
- Time to spread may be very long
  - Over-diagnosis (unnecessary treatment)
  - Patient may die of other causes

- What features characterize In Situ in older patients?
- What features change between older and younger?

Differential Prediction

Definition

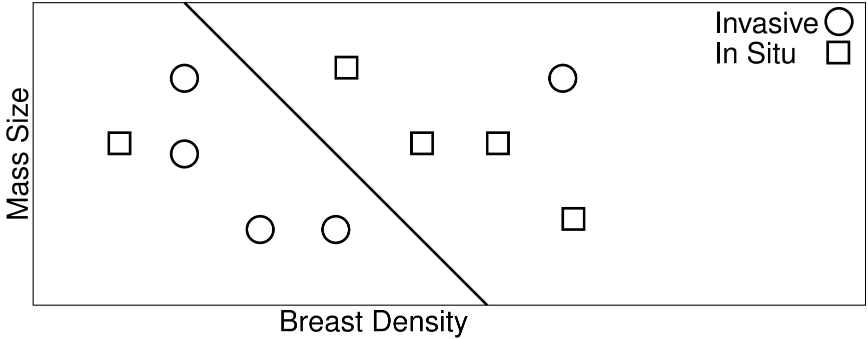
Differential Prediction (DP): Classifier exhibits significant performance differences over particular instance subgroups



Differential Prediction

Definition

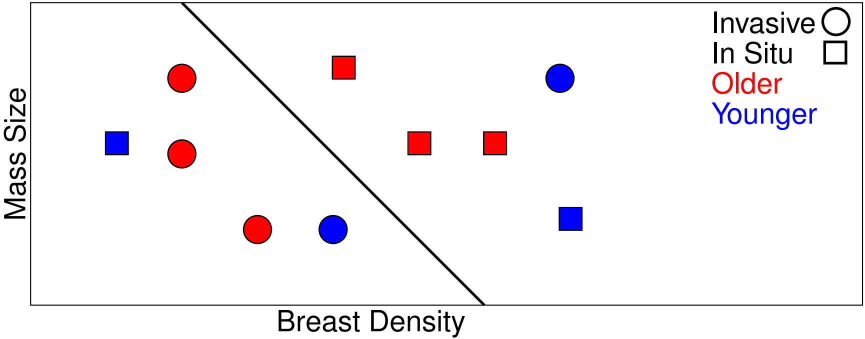
Differential Prediction (DP): Classifier exhibits significant performance differences over particular instance subgroups



Differential Prediction

Definition

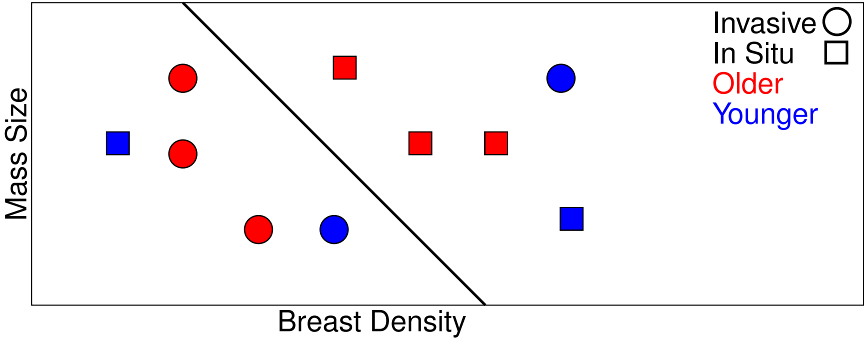
Differential Prediction (DP): Classifier exhibits significant performance differences over particular instance subgroups



Differential Prediction

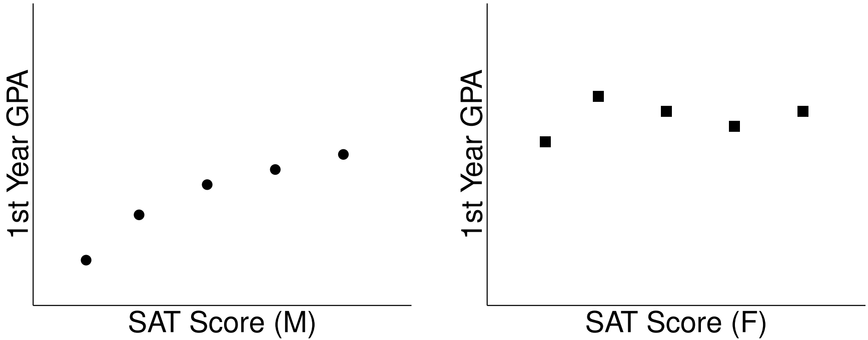
Definition

Differential Prediction (DP): Classifier exhibits significant performance differences over particular instance subgroups



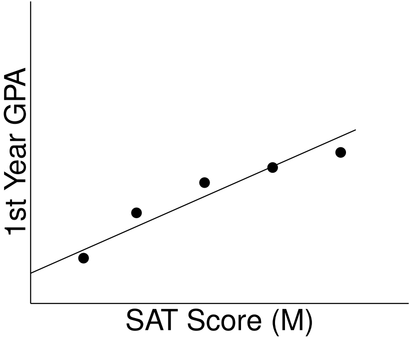
Using Regression to Detect DP

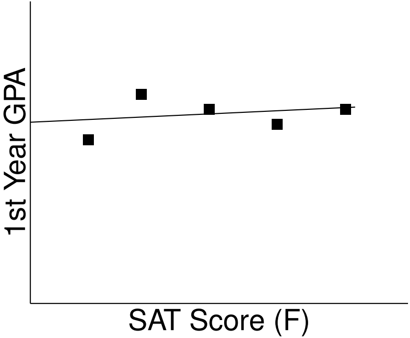
- Validate educational and psychological tests
- Detect discrepancies related to race or gender



Using Regression to Detect DP

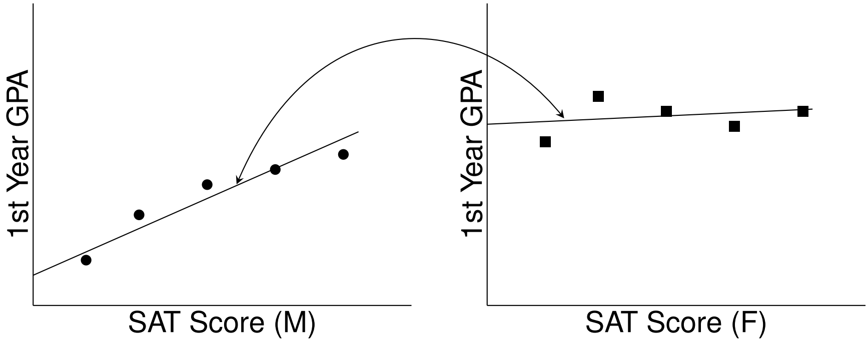
- Validate educational and psychological tests
- Detect discrepancies related to race or gender





Using Regression to Detect DP

- Validate educational and psychological tests
- Detect discrepancies related to race or gender



DP in Machine Learning

- Byproduct of classification
- Detected by:
  - Comparing classifiers built on distinct data subgroups
  - Checking classifier performance on multiple subgroups
- Differential misclassification cost: incorporating different misclassification costs into a cost sensitive classifier

Aim

- Classifier to maximize DP over specific data subsets
- Insight into DP features

DP in Machine Learning

- Byproduct of classification
- Detected by:
  - Comparing classifiers built on distinct data subgroups
  - Checking classifier performance on multiple subgroups
- Differential misclassification cost: incorporating different misclassification costs into a cost sensitive classifier

Aim

- Classifier to maximize DP over specific data subsets
- Insight into DP features

Outline

Inductive Logic Programming

Definition

Inductive Logic Programming (ILP): Machine learning approach that learns a set of first-order logic rules that explain the data

1. Generates easy to interpret if-then rules
2. Allows user interaction through background knowledge
3. Operates on relational datasets
4. Can investigate the performance of each rule, selecting for DP over given subsets

Inductive Logic Programming

Definition

Inductive Logic Programming (ILP): Machine learning approach that learns a set of first-order logic rules that explain the data

1. Generates easy to interpret if-then rules
2. Allows user interaction through background knowledge
3. Operates on relational datasets
4. Can investigate the performance of each rule, selecting for DP over given subsets

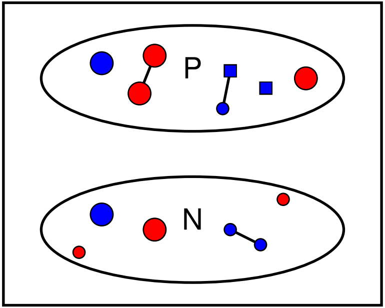
Inductive Logic Programming

Definition

Inductive Logic Programming (ILP): Machine learning approach that learns a set of first-order logic rules that explain the data

1. Generates easy to interpret if-then rules
2. Allows user interaction through background knowledge
3. Operates on relational datasets
4. Can investigate the performance of each rule, selecting for DP over given subsets

ILP Example

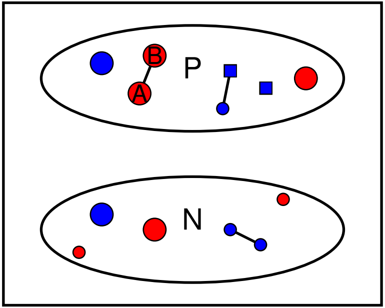


- $P(X)$ *if* $square(X)$
- $P(X)$ *if* $red(X) \wedge big(x)$
  - 1 false positive
- $P(X)$ *if* $sibling(X, Y) \wedge square(Y)$
- 1 false negative
- Form theory

Example

$P(A), red(A), big(A), round(A)$
$sibling(A, B)$

ILP Example

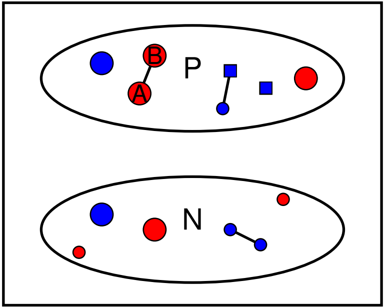


- $P(X)$ *if* $square(X)$
- $P(X)$ *if* $red(X) \wedge big(x)$
  - 1 false positive
- $P(X)$ *if* $sibling(X, Y) \wedge square(Y)$
- 1 false negative
- Form theory

Example

$P(A), red(A), big(A), round(A)$
$sibling(A, B)$

ILP Example

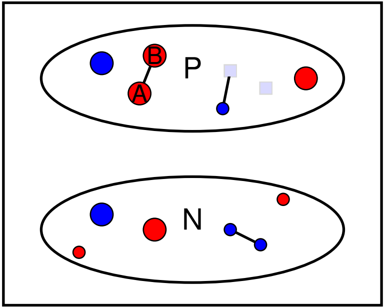


- $P(X)$ *if* $square(X)$
- $P(X)$ *if* $red(X) \wedge big(x)$
  - 1 false positive
- $P(X)$ *if* $sibling(X, Y) \wedge square(Y)$
- 1 false negative
- Form theory

Example

$P(A), red(A), big(A), round(A)$
$sibling(A, B)$

ILP Example

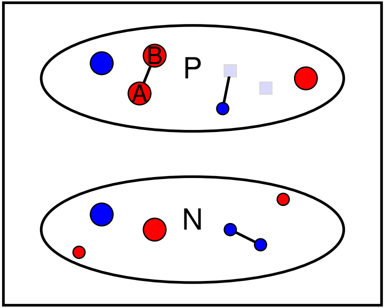


Example

$P(A), red(A), big(A), round(A)$
$sibling(A, B)$

- $P(X)$ *if* $square(X)$
- $P(X)$ *if* $red(X) \wedge big(x)$
  - 1 false positive
- $P(X)$ *if* $sibling(X, Y) \wedge square(Y)$
- 1 false negative
- Form theory

ILP Example

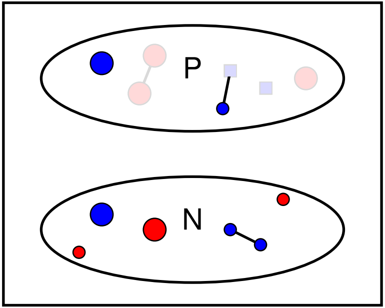


- $P(X)$ *if* $square(X)$
- $P(X)$ *if* $red(X) \wedge big(x)$
  - 1 false positive
- $P(X)$ *if* $sibling(X, Y) \wedge square(Y)$
- 1 false negative
- Form theory

Example

$P(A), red(A), big(A), round(A)$
$sibling(A, B)$

ILP Example

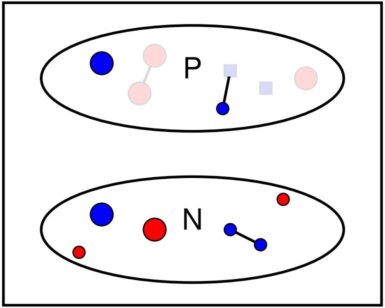


- $P(X)$ *if* $square(X)$
- $P(X)$ *if* $red(X) \wedge big(x)$
  - 1 false positive
- $P(X)$ *if* $sibling(X, Y) \wedge square(Y)$
- 1 false negative
- Form theory

Example

$P(A), red(A), big(A), round(A)$
$sibling(A, B)$

ILP Example

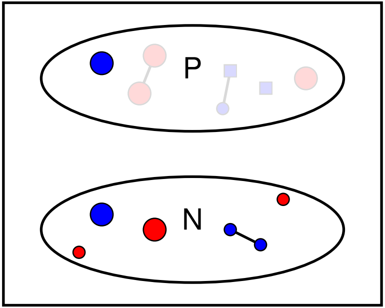


- $P(X)$ *if* $square(X)$
- $P(X)$ *if* $red(X) \wedge big(x)$
  - 1 false positive
- $P(X)$ *if* $sibling(X, Y) \wedge square(Y)$
- 1 false negative
- Form theory

Example

$P(A), red(A), big(A), round(A)$
$sibling(A, B)$

ILP Example

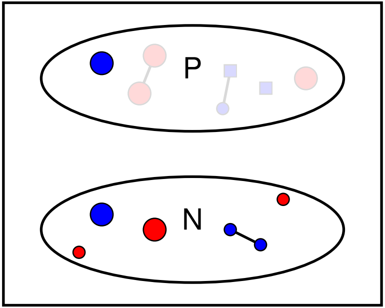


- $P(X)$ *if* $square(X)$
- $P(X)$ *if* $red(X) \wedge big(x)$
  - 1 false positive
- $P(X)$ *if* $sibling(X, Y) \wedge square(Y)$
- 1 false negative
- Form theory

Example

$P(A), red(A), big(A), round(A)$
$sibling(A, B)$

ILP Example



- $P(X)$ *if* $square(X)$
- $P(X)$ *if* $red(X) \wedge big(x)$
  - 1 false positive
- $P(X)$ *if* $sibling(X, Y) \wedge square(Y)$
- 1 false negative
- Form theory

Example

$P(A), red(A), big(A), round(A)$
$sibling(A, B)$

Outline

Breast-Cancer Stage Modeling

- Identify patient subgroups that would benefit most from treatment
- Invasive and In Situ characteristics in older and younger women
- Data is mostly in free-text

Tasks

- DP features for Invasive and In Situ
- Information extraction from free-text

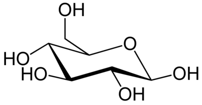
Breast-Cancer Stage Modeling

- Identify patient subgroups that would benefit most from treatment
- Invasive and In Situ characteristics in older and younger women
- Data is mostly in free-text

Tasks

- DP features for Invasive and In Situ
- Information extraction from free-text

Hexose-Binding Modeling

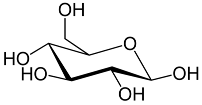


- Galactose, glucose, mannose
- High specificity to diverse protein families
- Interesting to uncover differential binding patterns

Tasks

- Glucose-binding model
- Data-driven empirical validation of biochemical findings

Hexose-Binding Modeling



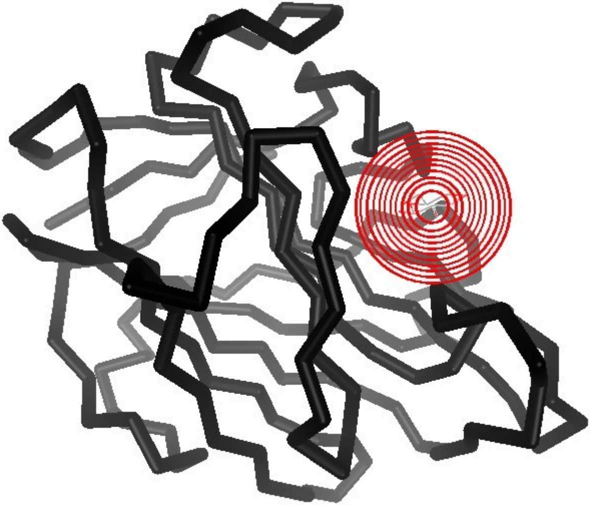
- Galactose, glucose, mannose
- High specificity to diverse protein families
- Interesting to uncover differential binding patterns

Tasks

- Glucose-binding model
- Data-driven empirical validation of biochemical findings

Outline

Hexose Binding-Site Representation



Hexose Binding-Site Features

```
procedure EXTRACTFEATURES(binding site center)
    for all concentric layers do
        for all PDB atoms do
            get distance from center
            get charge
            get hydrophobicity
            get hydrogen-bonding
            get residue
        end for
    end for
end procedure
```

Glucose Binding-Site Classifier (*Proteins*)

- Random Forests for feature selection
- Support Vector Machines for classification

Features	L1	L2	L3	L4	L5	L6	L7	L8
Negative Charge			X				X	X
Neutral Charge	X	X						
Non H-Bonding	X							
H-Bonding	X		X					X
Hydrophilic	X		X					X
Hydroneutral		X	X					
Hydrophobic					X		X	
Neutral Residue				X	X		X	
Acidic Residue			X		X	X	X	X

Validating Hexose-Binding Knowledge (*ILP'09*)

- Use ILP system Aleph
- Extract rules from data without prior biochemical knowledge
- Compare resulting rules with known biochemical rules
- Induce most of the known hexose-binding biochemical rules
- Find a previously unreported dependency between TRP and GLU

Validating Hexose-Binding Knowledge (*ILP'09*)

- Use ILP system Aleph
- Extract rules from data without prior biochemical knowledge
- Compare resulting rules with known biochemical rules
- Induce most of the known hexose-binding biochemical rules
- Find a previously unreported dependency between TRP and GLU

Outline

Breast-Cancer Stages

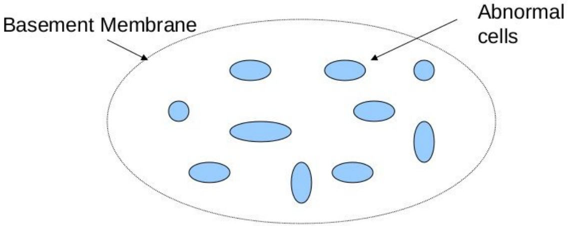


Figure: In-Situ Cancer Stage

Breast-Cancer Stages

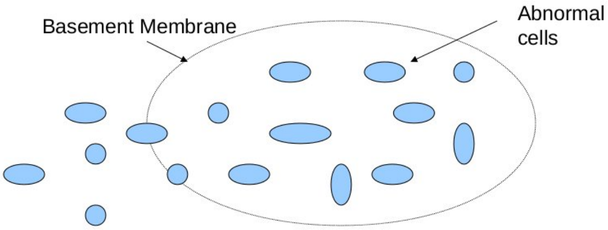


Figure: Invasive Cancer Stage

Age Matters

- Apply linear logistic regression
- Uncover a differential ability in predicting invasive and in-situ cancer in older vs. younger women
- Stratify our data:

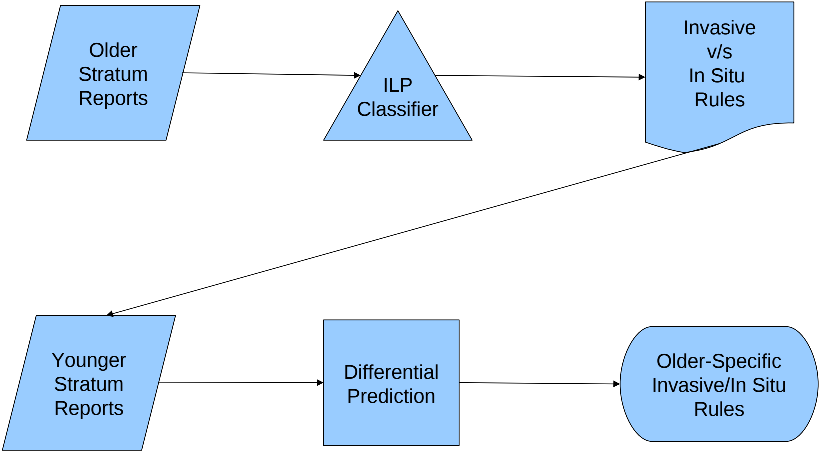
  Younger: $<$ 50 years, pre-menopausal
  Middle: $[50, 65)$ years, peri-menopausal
  Older: $>=$ 65 years, post-menopausal

Age Matters

- Apply linear logistic regression
- Uncover a differential ability in predicting invasive and in-situ cancer in older vs. younger women
- Stratify our data:

  Younger: $<$ 50 years, pre-menopausal
  Middle: $[50, 65)$ years, peri-menopausal
  Older: $>=$ 65 years, post-menopausal

Generate-then-Test DP Method (*IHI'10*)



Middle-Cohort Precision Comparison

	Comparing Middle Cohort with:	
Rule	Older Cohort (*p*-value)	Younger Cohort (*p*-value)
Invasive Older Prediction		
Rule 1	0.04*	0.50
Rule 2	0.01*	0.32
Rule 3	0.05	0.49
Rule 4	0.26	0.00*
Rule 5	0.48	0.00*
In-Situ Older Prediction		
Rule 1	0.27	0.06
Invasive Younger Prediction		
Rule 1	0.00*	0.12
In-Situ Younger Prediction		
Rule 1	0.10	0.06

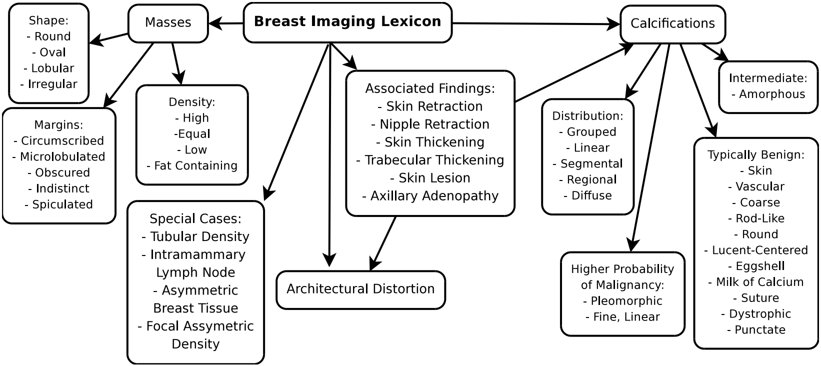
* Statistically significant at the 95% confidence level.

Mammography Features

Structured	Extracted using NLP
Family breast cancer history	Mass margin
Personal breast cancer history	Mass shape
Prior surgery	Calcification distribution
Palpable lump	Calcification morphology
Screening v/s diagnostic	Architectural distortion
Indication for exam	Associated findings
Breast Density	Mammary lymph node
BI-RADS code left	Asymmetric breast tissue
BI-RADS code right	Focal asymmetric density
BI-RADS code combined	Tubular density
Principal finding	Mass size

Outline

Breast Imaging Reporting & Data System (BI-RADS)

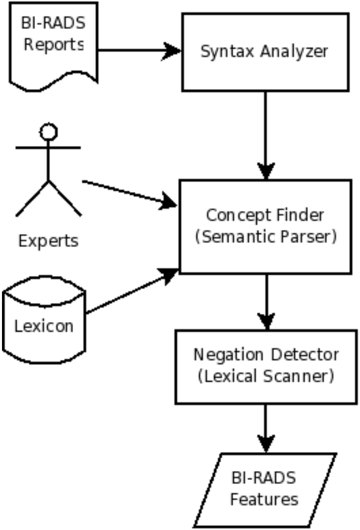


Information from Lexicon

- Lexicon specifies synonyms
  - E.g.: Equal density, Isodense
- Lexicon allows for ambiguous wording

Text	Concept
indistinct margin	indistinct margin
indistinct calcification	amorphous calcification
indistinct image	not a BI-RADS concept

Algorithm Flowchart (*ICDM-W'09*)



- Context Free Grammar
- Straight-forward negation
- Negation-deactivation triggers

Rule Generation Example

- Aim: Skin Thickening concept
- Lexicon specifies “skin thickening”
- Try “skin” and “thickening” in same sentence
  - thickening of the overlying skin
  - marker placed on the skin overlying a palpable focal area of thickening in the upper outer right breast
- Experts suggest “skin” and “thickening” in close proximity
- Start with a large scope
  - Assess number of true and false positives
- Move to smaller scopes
  - Assess number of false negatives
- Experts decide on the best distance

Rule Generation Example

- Aim: Skin Thickening concept
- Lexicon specifies “skin thickening”
- Try “skin” and “thickening” in same sentence
  - thickening of the overlying skin
  - marker placed on the skin overlying a palpable focal area of thickening in the upper outer right breast
- Experts suggest “skin” and “thickening” in close proximity

- Start with a large scope
  - Assess number of true and false positives
- Move to smaller scopes
  - Assess number of false negatives
- Experts decide on the best distance

Outline

DP Rules Generation Paradigm

Aim

Formally define the differential predictive rules generation paradigm

Definition

DP Rule/Concept: Given a stratified dataset, a rule/concept whose performance is significantly better over one stratum as compared to the others

DP Rules Generation Paradigm

Aim

Formally define the differential predictive rules generation paradigm

Definition

DP Rule/Concept: Given a stratified dataset, a rule/concept whose performance is significantly better over one stratum as compared to the others

K-Stratified Dataset

Definition (Stratified Dataset)

Let c be a concept defined over the set of instances X, and let $D = \{\langle x, c(x) \rangle\}$ be a set of training examples labeled according to c. Let D_i be Q disjoint subsets of D, with $Q \geq 2$, and let D_i^l be the training examples of D_i that have class label l, such that:

$$(\forall (i, j) \in [1, Q],\ i \neq j)\ D_i \subset D,\ D_i \cap D_j = \emptyset,\ \forall l\ D_i^l \neq \emptyset. \qquad (1)$$

A **K-stratified dataset** $\mathscr{D}$ over the set of instances X is the union of K such subsets D_i, with $2 \leq K \leq Q$, such that:

$$\mathscr{D} = \{D_i \mid 1 \leq i \leq K\}. \qquad (2)$$

Differential Predictive Concept

Definition (Differential Predictive Concept)

Let c be a concept over the set of instances X, and let $\mathscr{D}$ be a K-stratified dataset. Let $S(c, D_i)$ be the classification performance score for c over the subset D_i. A **stratum-j specific differential predictive concept** is a concept c_j such that:

$$S(c_j, D_j) \gg S(c_j, D_i), \ (\forall i \neq j). \tag{3}$$

- The score difference can be evaluated using statistical significance tests or by setting a threshold

Outline

1. Motivation
   - Differential Prediction (DP)
   - Inductive Logic Programming (ILP)
   - Applications
2. Preliminary Results
   - Predicting Hexose Binding Sites
   - DP for Invasive/In-Situ
   - BI-RADS Information Extraction
3. **Proposed Work**
   - Differential Predictive Rules Definition
   - **DP within the ILP Framework**
   - Randomizing Recall
   - BI-RADS Terms Annotation
4. Wrap-Up

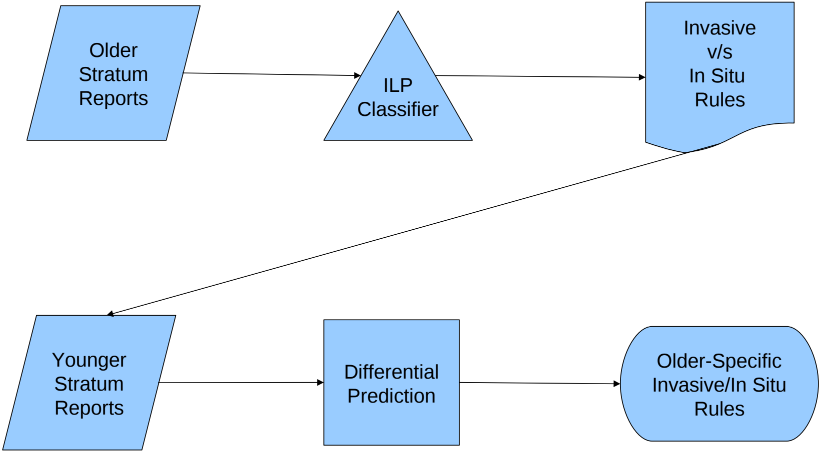
DP within the ILP Framework

Aim

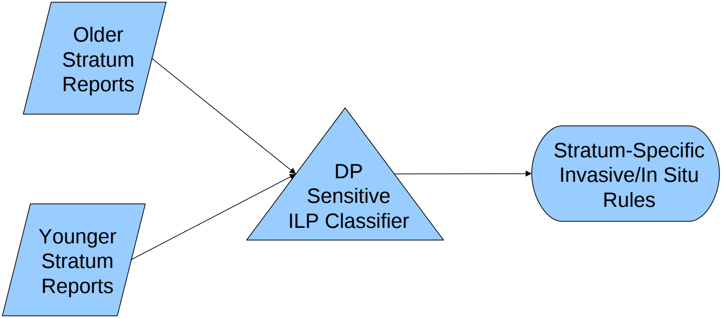
Implement DP rules generation within ILP

- Generate-then-test approach
- Test-incorporation approach, more rigorous
- Alter the ILP search
- Alter evaluation function to score a clause according to its DP performance over stratified training set
- Return rules selected for their DP score

Generate-then-Test DP Method (*IHI'10*)



Test-Incorporation DP Method



DP-Sensitive Scoring Function

Definition (DP-Sensitive Scoring Function)

Let R be a clause over the set of instances X, and let $\mathscr{D}$ be a 2-stratified dataset over X. Let $S(R, D_i)$ be the classification performance score for R over the subset D_i. We define the **differential-prediction-sensitive scoring function** Q as

$$Q(R, D_1, D_2) = S(R, D_1) - S(R, D_2). \tag{4}$$

Advantages

- Any classification scoring function S can be used
- Generates a set of rules as a consistent theory

DP-Sensitive Scoring Function

Definition (DP-Sensitive Scoring Function)

Let R be a clause over the set of instances X, and let $\mathscr{D}$ be a 2-stratified dataset over X. Let $S(R, D_i)$ be the classification performance score for R over the subset D_i. We define the **differential-prediction-sensitive scoring function** Q as

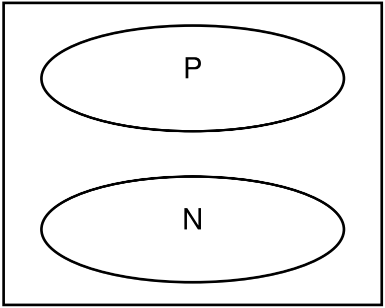
$$Q(R, D_1, D_2) = S(R, D_1) - S(R, D_2). \tag{4}$$

Advantages

- Any classification scoring function S can be used
- Generates a set of rules as a consistent theory

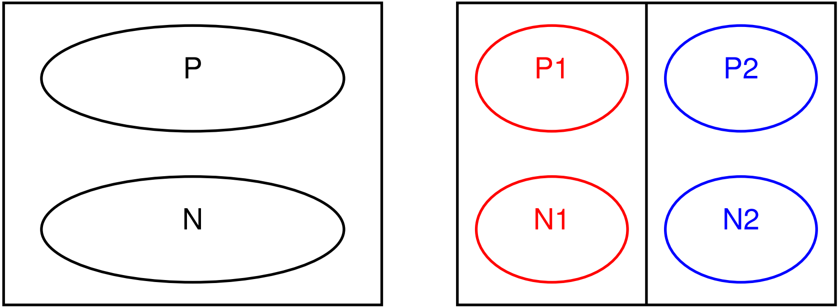
Coverage Scoring Function

- Rule coverage score: $Cover(P) - Cover(N)$
- DP: $(Cover(P1) - Cover(N1)) - (Cover(P2) - Cover(N2))$

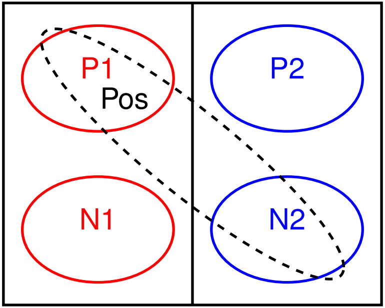


Coverage Scoring Function

- Rule coverage score: $Cover(P) - Cover(N)$
- DP: $(Cover(P1) - Cover(N1)) - (Cover(P2) - Cover(N2))$

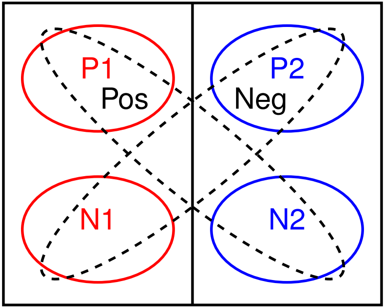


Instance Relabeling DP Method



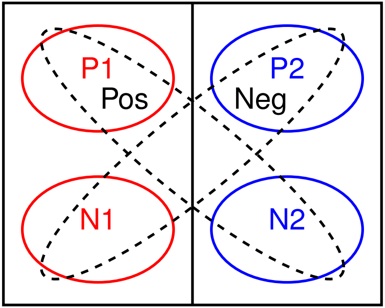
- Relabel $Pos = P1 + N2$
- Relabel $Neg = P2 + N1$
- Run standard ILP
- $Cover(Pos) - Cover(Neg)$
- $Cover(P1+N2) - Cover(P2+N1)$
- $(Cover(P1) + Cover(N2)) - (Cover(P2) + Cover(N1))$
- $(Cover(P1) - Cover(N1)) - (Cover(P2) - Cover(N2))$

Instance Relabeling DP Method



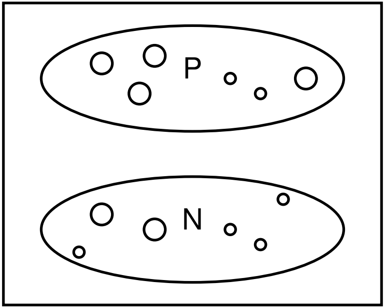
- Relabel $Pos = P1 + N2$
- Relabel $Neg = P2 + N1$
- Run standard ILP
- $Cover(Pos) - Cover(Neg)$
- $Cover(P1+N2) - Cover(P2+N1)$
- $(Cover(P1) + Cover(N2)) - (Cover(P2) + Cover(N1))$
- $(Cover(P1) - Cover(N1)) - (Cover(P2) - Cover(N2))$

Instance Relabeling DP Method



- Relabel $Pos = P1 + N2$
- Relabel $Neg = P2 + N1$
- Run standard ILP
- $Cover(Pos) - Cover(Neg)$
- $Cover(P1+N2) - Cover(P2+N1)$
- $(Cover(P1) + Cover(N2)) - (Cover(P2) + Cover(N1))$
- $(Cover(P1) - Cover(N1)) - (Cover(P2) - Cover(N2))$

Baseline DP Method

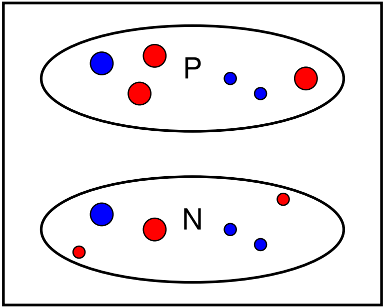


- Include stratifying attribute as a predicate *p*
- Run ILP over whole dataset
- Select rules containing the predicate *p*
- Rules specific to the stratum the predicate *p* refers to

Example

$P(X)$ *if* $red(X) \wedge big(X)$

Baseline DP Method

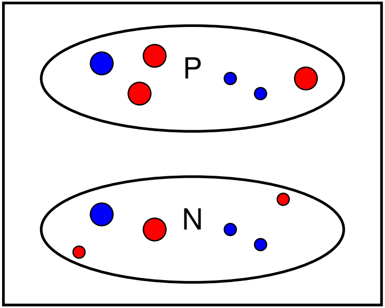


- Include stratifying attribute as a predicate p
- Run ILP over whole dataset
- Select rules containing the predicate p
- Rules specific to the stratum the predicate p refers to

Example

$P(X)$ *if* $red(X) \wedge big(X)$

Baseline DP Method



- Include stratifying attribute as a predicate p
- Run ILP over whole dataset
- Select rules containing the predicate p
- Rules specific to the stratum the predicate p refers to

Example

$P(X)$ *if* $red(X) \wedge big(X)$

Implementing K-Stratified DP

- Reduce a K-strata problem to K 2-strata problems
- Keep stratum i, collapse others together
- Extract stratum i DP rules

- Multi-strata DP-sensitive scoring function
- f-divergence functions?

Implementing K-Stratified DP

- Reduce a K-strata problem to K 2-strata problems
- Keep stratum i, collapse others together
- Extract stratum i DP rules

- Multi-strata DP-sensitive scoring function
- f-divergence functions?

Outline

Aleph (Top-Down)

Require: Examples E, mode declarations M, background knowledge B, Scoring function S

Learned_rules $\leftarrow \{\}$
Pos $\leftarrow$ all positive examples in *E*
while *Pos* **do**
    Select example $e \in Pos$
    Construct bottom clause $\perp_e$ from *e*, *M* and *B*   ▷ Saturation step
    Candidate_literals $\leftarrow$ *Literals*($\perp_e$)
    New_rule $\leftarrow$ *pos*(**X**)   ▷ Most general rule
    repeat   ▷ Top-down reduction step
        Best_literal $\leftarrow \underset{L \in Candidate\_literals}{\text{argmax}}$ S(*New_rule* with precondition *L*)
        Add *Best_literal* to preconditions of *New_rule*
    until No more *S*(*New_rule*) score improvement
    Learned_rules $\leftarrow$ *Learned_rules* $+$ *New_rule*
    Pos $\leftarrow$ *Pos* $-$ {members of *Pos* covered by *New_rule*}
end while
return *Learned_rules*

ProGolem (Bottom-Up)

Require: Examples E, mode declarations M, background knowledge B, Scoring function S

Learned_rules $\leftarrow \{\}$
Pos $\leftarrow$ all positive examples in *E*
while *Pos* **do**
    Select example $e \in Pos$
    Construct bottom clause $\perp_e$ from *e*, *M* and *B* ▷ Saturation step
    New_rule $\leftarrow \perp_e$ ▷ Most specific rule
    repeat ▷ Bottom-up reduction step
        Select a different example $e' \in Pos$
        Blocking_literals $\leftarrow$ *ARMG*(*New_rule*, e')
        Remove *Blocking_literals* from preconditions of *New_rule*
    until No more *S*(*New_rule*) score improvement
    Learned_rules $\leftarrow$ *Learned_rules* $+$ *New_rule*
    Pos $\leftarrow$ *Pos* $-$ {members of *Pos* covered by *New_rule*}
end while
return *Learned_rules*

Bottom-Up Search Advantages

- Omitted Variable Problem
- Not considering a DP variable
- Bottom-up starts with all attributes

- Myopia Effect
- Top-down search assumes literals conditionally independent given target class
- If features highly correlated, searches very similar hypotheses

Bottom-Up Search Advantages

- Omitted Variable Problem
- Not considering a DP variable
- Bottom-up starts with all attributes

- Myopia Effect
- Top-down search assumes literals conditionally independent given target class
- If features highly correlated, searches very similar hypotheses

Non-Determinacy and Recall

Example

$legalName(Joe, X); parent(Joe, Y); sibling(Joe, Z)$

Definition

Predicate Non-Determinacy: The number of possible solutions of a given predicate

Determinate Predicate: At most one solution

Definition

Recall: Imposed bound on predicate non-determinacy

Non-Determinacy and Recall

Example

$legalName(Joe, X); parent(Joe, Y); sibling(Joe, Z)$

Definition

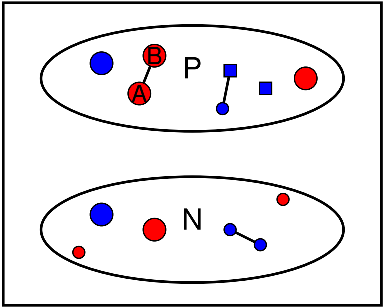
Predicate Non-Determinacy: The number of possible solutions of a given predicate

Determinate Predicate: At most one solution

Definition

Recall: Imposed bound on predicate non-determinacy

Randomized ProGolem



Example (Bottom Clause (A))

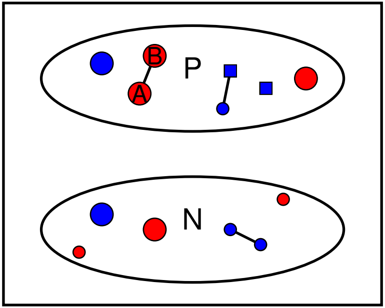
red(*A*), *big*(*A*), *round*(*A*),
sibling(*A*, *B*),
red(*B*), *big*(*B*), *round*(*B*)

- Highly non-determinate data
- Exponential learning time for bottom-up learner
- ProGolem: limit bottom clause to first *recall* instantiations

Aim

- Randomize ProGolem recall
- Use it for DP

Randomized ProGolem



- Highly non-determinate data
- Exponential learning time for bottom-up learner
- ProGolem: limit bottom clause to first *recall* instantiations

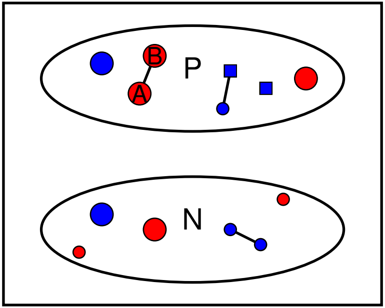
Aim

- Randomize ProGolem recall
- Use it for DP

Example (Bottom Clause (A))

red(*A*), *big*(*A*), *round*(*A*),
sibling(*A*, *B*),
red(*B*), *big*(*B*), *round*(*B*)

Randomized ProGolem



- Highly non-determinate data
- Exponential learning time for bottom-up learner
- ProGolem: limit bottom clause to first *recall* instantiations

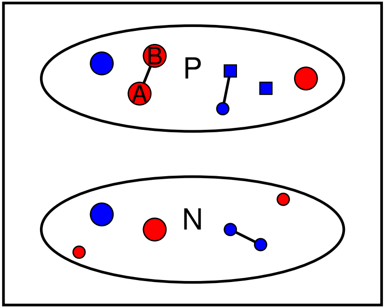
Aim

- Randomize ProGolem recall
- Use it for DP

Example (Bottom Clause (A))

$red(A), big(A), round(A),$
$sibling(A, B),$
$red(B), big(B), round(B)$

Randomized ProGolem



- Highly non-determinate data
- Exponential learning time for bottom-up learner
- ProGolem: limit bottom clause to first *recall* instantiations

Example (Bottom Clause (A))

$red(A), big(A), round(A),$
$sibling(A, B),$
$red(B), big(B), round(B)$

Aim

- Randomize ProGolem recall
- Use it for DP

Outline

BI-RADS Terms Annotation

Aim

Improve BI-RADS extraction from free-text

- Current method maps words to concepts
- Extend to term annotation
  - Create first BI-RADS annotation tool
  - Attempt new term/concept discovery
- Transfer method to other languages (Portuguese)

BI-RADS Terms Annotation

Aim

Improve BI-RADS extraction from free-text

- Current method maps words to concepts
- Extend to term annotation
  - Create first BI-RADS annotation tool
  - Attempt new term/concept discovery
- Transfer method to other languages (Portuguese)

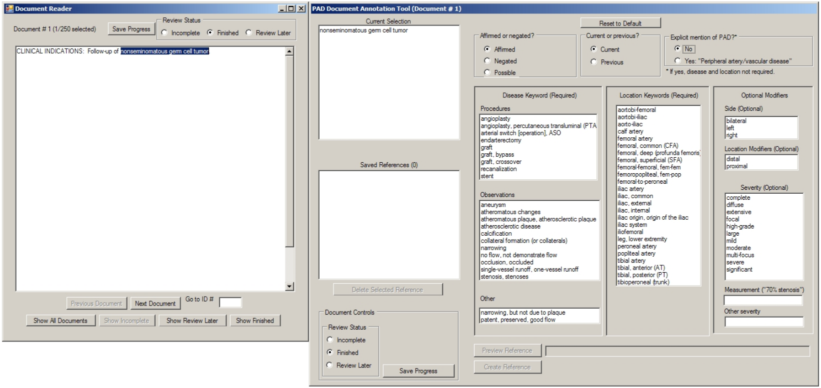
BI-RADS Terms Annotation

Aim

Improve BI-RADS extraction from free-text

- Current method maps words to concepts
- Extend to term annotation
  - Create first BI-RADS annotation tool
  - Attempt new term/concept discovery
- Transfer method to other languages (Portuguese)

BI-RADS Annotator Template



Outline

1. Motivation
   - Differential Prediction (DP)
   - Inductive Logic Programming (ILP)
   - Applications
2. Preliminary Results
   - Predicting Hexose Binding Sites
   - DP for Invasive/In-Situ
   - BI-RADS Information Extraction
3. Proposed Work
   - Differential Predictive Rules Definition
   - DP within the ILP Framework
   - Randomizing Recall
   - BI-RADS Terms Annotation
4. Wrap-Up

Timeline

Fall 2010	Formally define DP rules Translate rules into Portuguese
Spring 2011	Randomize and test ProGolem recall Implement BI-RADS annotator
Fall 2011	Implement and test ILP-based DP methods Extract breast cancer DP rules
Spring 2012	Wrap-up work Write and defend thesis

Bibliography

H. Nassif, H. Al-Ali, S. Khuri, and W. Keyrouz.
Prediction of Protein-Glucose Binding Sites Using SVMs.
Proteins, 77(1):121-132, 2009.

H. Nassif, D. Page, M. Ayvaci, J. Shavlik, and E.S. Burnside.
Uncovering Age-Specific Invasive and DCIS Breast Cancer Rules Using ILP.
IHI'10, Arlington, VA, pp. 76-82, 2010.

H. Nassif, H. Al-Ali, S. Khuri, W. Keyrouz and D. Page.
An ILP Approach to Validate Hexose Binding Biochemical Knowledge.
ILP'09, Leuven, Belgium, pp. 149-165, 2009.

H. Nassif, R. Wood, E.S. Burnside, M. Ayvaci, J. Shavlik and D. Page.
Information Extraction for Clinical Data Mining: A Mammography Case Study
ICDM-Workshop'09, Miami, pp. 37-42, 2009.

Summary

- First glucose-binding model
- Validate hexose-binding knowledge
- BI-RADS extractor
- First DP rules generation

- Formally define DP rules generation paradigm
- Implement DP rules within ILP
- Randomize ProGolem recall
- Improve BI-RADS extraction from free-text

Hexose Features

Atomic Feature	Values
Charge	Negative, Neutral, Positive
Hydrogen-bonding	Non-hydrogen bonding, Hydrogen-bonding
Hydrophobicity	Hydrophilic, Hydroneutral, Hydrophobic

Residue Grouping	Amino Acids
Aromatic	HIS, PHE, TRP, TYR
Aliphatic	ALA, ILE, LEU, MET, VAL
Neutral	ASN, CYS, GLN, GLY, PRO, SER, THR
Acidic	ASP, GLU
Basic	ARG, LYS

Atomic Chemical Properties I

PDB atom symbol	Residues	Partial Charge	Hydro-phobicity	Hydrogen Bonding
Amino acid oxygen atoms				
O	All amino acids	0	HPHIL	HB
OXT	All amino acids	-ve	HPHIL	HB
OE1, OE2, OD1, OD2	GLU, ASP	-ve	HPHIL	HB
OE1, OD1	GLN, ASN	0	HPHIL	HB
OG, OG1, OH	SER, THR, TYR	0	HPHIL	HB
Amino acid carbon atoms				
C	All amino acids	0	HNEUT	NHB
CA	All amino acids	0	HNEUT	NHB
CB, CG, CD, CE	ALA, SER, THR, CYS, ASP, ASN, GLU, GLN, ARG, LYS, PRO	0	HNEUT	NHB
CB, CG, CD, CE	LEU, VAL, ILE, MET	0	HPHOB	NHB
CG1, CG2, CD1, CD2, CD1	LEU, VAL, ILE	0	HPHOB	NHB
CG, CD1, CD2, CE1, CE2, CZ, CG,CD1, CD2, CE2, CE3, CZ2, CZ3, CH2	PHE, TYR, TRP	0	HPHOB	NHB
CG, CD2, CE1	HIS	0	HPHOB	NHB

Atomic Chemical Properties II

PDB atom symbol	Residues	Partial Charge	Hydro-phobicity	Hydrogen Bonding
Amino acid nitrogen atoms				
N	All amino acids except PRO	0	HPHIL	HB
N	PRO	0	HPHIL	NHB
NE2, ND2	GLN, ASN	0	HPHIL	HB
NZ	LYS	+ve	HPHIL	HB
NE	ARG	+ve	HPHIL	NHB
NH1, NH2	ARG	+ve	HPHIL	HB
ND1, NE2	HIS	0	HPHIL	HB
NE1	TRP	0	HNEUT	NHB
Amino acid sulfur atoms				
SG	CYS	0	HPHIL	HB
SD	MET	0	HNEUT	NHB
Water and ions atoms				
O	HOH	0	HPHIL	HB
O1, O2, O3, O4	SO4, 2HP	-ve	HPHIL	HB
CA, MG, ZN	CA, MG, ZN	+ve	HPHIL	HB

SVM and RF Results

Property	RF	Feature Number	Error (%)	Sensitivity (%)	Specificity (%)	Support Vectors (%)
Charge	false	24	24.32	79.31	73.33	77.03
	true	5	14.86	86.21	84.44	44.59
Hydrogen Bonding	false	16	17.57	82.76	82.22	41.89
	true	3	14.86	82.76	86.67	47.30
Hydro-phobicity	false	24	16.22	72.41	91.11	65.57
	true	15	12.16	82.76	91.11	40.54
Residue Grouping	false	48	21.62	48.28	97.78	100.0
	true	19	09.46	93.10	88.89	41.89
Features Combined	false	112	18.92	75.86	84.44	79.73
	true	24	08.11	89.66	93.33	40.54

Age Cohorts

Subset	Invasive	In-Situ	Subset Total
Younger1	132	55	187
Younger2	132	55	187
Younger Total	264	110	374
Middle1	199	85	284
Middle2	199	85	284
Middle Total	398	170	568
Older1	200	66	266
Older2	201	66	267
Older Total	401	132	533
Grand Total	1063	412	1475

Comparing Automated and Manual Extraction

- Automated method superior to manual method ($p = 0.024$)
- Probabilistic interpretation of F-score with Laplace prior

		Actual	
Method	Predicted	Feature Present	Absent
Automated	Feature Present	211	5
	Feature Absent	10	4074
Manual	Feature Present	198	5
	Feature Absent	23	4074