Centre for
Data Analytics

Insight

Efficient Sequence Regression by Learning Linear Models in All-Subsequence Space

Severin Gsponer, Barry Smyth, Georgiana Ifrim

19.09.2017

Outline

- Sequence Regression Problem
- Related Work and Background
- Our Approach: Linear Model for Sequence Regression (SqLoss)
- Evaluation

Sequence Regression Problem

Problem Setting

Score	Sequence
290.5	AGTCCACAAGGCTAGGATAGCTATCCGGATCGA
315.1	TATCCTGCAGTACAAGTCCGTAATTCTCAATCCA
805.6	AGTCCGCTAGGCTAGGATAGCTAGCCCGATCGA
799.7	AGCCAAGACCTGAAATAGGCTCCTGAGATACAG
???	CGGGTCGTATCCGCACTGAATATCCAGAGATACG

$$\Sigma = \{A, C, G, T\}$$

Sequence Regression Problem

Problem Setting

Score	Sequence
290.5	AGTCC*C*AGGCTAGGATAGCTATCCGGATCGA
315.1	TATCCTGCAGTACAAGTCCGTAATTC*C*ATCCA
805.6	AGTCCGCTAGGCTAGGATAGCTAGCCCGATCGA
799.7	AGCCAAGACCTGAAATAGGCTCCTGAGATACAG
???	CGGGTCGTATCCGCACTGAATATCTAGGCTTACG

$$\Sigma = \{A, C, G, T\}$$

Weight	k-mer
796.6	TAGGCT
402,5	C*C*A
-125.3	TCCG

Related Work

Bag of Words

- Loss of structural order (e.g., Mary is faster than John)
- Often not accurate enough

Kernel SVM

- Raise into implicit high-dimensional feature space through kernel trick
- Restrict features for scale (e.g., max 5-mer)
- Not easily interpretable (Blackbox)

Neural Networks

- Hard to train
- Time consuming
- Not easily interpretable (Blackbox)

Design Goals

Interpretable and Simple

Linear models with k-mers as features

Accurate

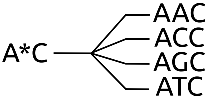
All-length k-mer feature space and wildcards

Efficient

Fast search for most promising features

All-Subsequence Feature Space

k-mers	A	C	G	T	AA	AC	...	CA	CC	...	GTCCTA	GTCCTC	...	TTTTTT
Binary vector	1	1	1	1	1	0	...	0	1	...	1	0	...	0



Sample sequence:	...GTCCTAATCCTA ...	
1-mer:	A, C, G, T	(4 possible)
2-mer:	GT, TC, CC, CT, TA, ...	($4^2 = 16$ possible)
3-mer:	GTC, TCC, CCT, CTA, TAA, ...	($4^3 = 64$ possible)
$\vdots$	$\vdots$	
8-mer:	GTCCTAAT, TCCTAATC, ...	($4^8 = 65536$ possible)

Regression with Square Loss

Given:

Training set of labeled examples:

$$\{x_i, y_i\} \text{ for } i = 1, \dots, N \quad \text{where } y_i \in \mathbb{R}$$

$$\boldsymbol{x_i} \in \mathbb{R}^d \quad \text{with } d = \text{number of features}$$

Goal:

Find $\boldsymbol{\beta} = (\beta_1, \beta_2, \dots, \beta_d)$, $\beta_i \in \mathbb{R}$ by optimizing:

$$\beta^* = \underset{\beta \in \mathbb{R}^d}{\arg\min}\, L(\beta) = \underset{\beta \in \mathbb{R}^d}{\arg\min} \sum_{i=1}^{N} (y_i - \boldsymbol{\beta}^T x_i)^2 + CR(\beta)$$

Classical gradient descent is computationally infeasible for a large feature space

$$\boldsymbol{\beta}^{(t)} = \boldsymbol{\beta}^{(t-1)} - \eta_t \boldsymbol{\nabla L}(\boldsymbol{\beta}^{(\boldsymbol{t-1})})$$

Our Approach: SqLoss Algorithm

Algorithm 1 Coordinate Descent with Gauss Southwell Selection

```
Set β^(0) = 0
while not termination condition do
    Adjust intercept
    Calculate objective function L(β^(t))
    Find coordinate j_t with maximum gradient value
    Find optimal step size η_{j_t} by line search or exact optimization
    Update β^(t) = β^(t−1) − η_{j_t} ∂L/∂β_{j_t}(β^(t−1)) e_{j_t}
    Add corresponding feature to feature set
end while
```

Our Approach: SqLoss Algorithm

Algorithm 1 Coordinate Descent with Gauss Southwell Selection

Set $\beta^{(0)} = 0$
while not termination condition **do**
    Adjust intercept
    Calculate objective function $L(\beta^{(t)})$
    Find coordinate j_t with maximum gradient value
    Find optimal step size η_{j_t} by line search or exact optimization
    Update $\beta^{(t)} = \beta^{(t-1)} - \eta_{j_t} \frac{\partial L}{\partial \beta_{j_t}}(\beta^{(t-1)}) e_{j_t}$
    Add corresponding feature to feature set
end while

How do we find coordinate j_t efficiently?

Efficient GS Selection via Gradient Bounding

Example

$$s_j = "ACTC" \text{ and } s_p = "ACT"$$
$$gradient(s_j) \leq \mu(s_p)$$

Key Ideas

Bound gradient of each feature using only information about its prefix.

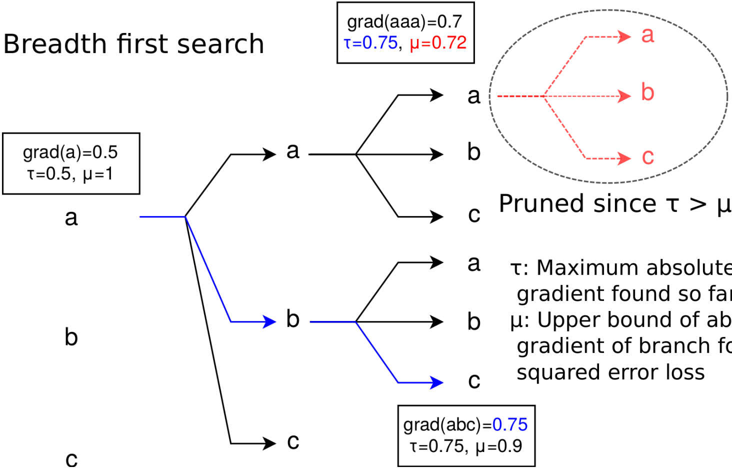
Separate positive and negative terms of the gradient.

Theorem For any subsequence $s_p \subseteq s_j$ it holds:

$$\left|\frac{\partial L}{\partial \beta_j}(\beta)\right| \leq \mu(s_p) = \max\{\left|\sum_{\{i|s_p \in x_i, y_i - \beta^T x_i \geq 0\}} -2 \cdot x_{ip} \cdot (y_i - \beta^T \cdot x_i)\right|,$$
$$\left|\sum_{\{i|s_p \in x_i, y_i - \beta^T x_i \leq 0\}} -2 \cdot x_{ip} \cdot (y_i - \beta^T \cdot x_i)\right|\}$$

Search and Pruning

Breadth first search



τ: Maximum absolute gradient found so far

μ: Upper bound of absolute gradient of branch for squared error loss

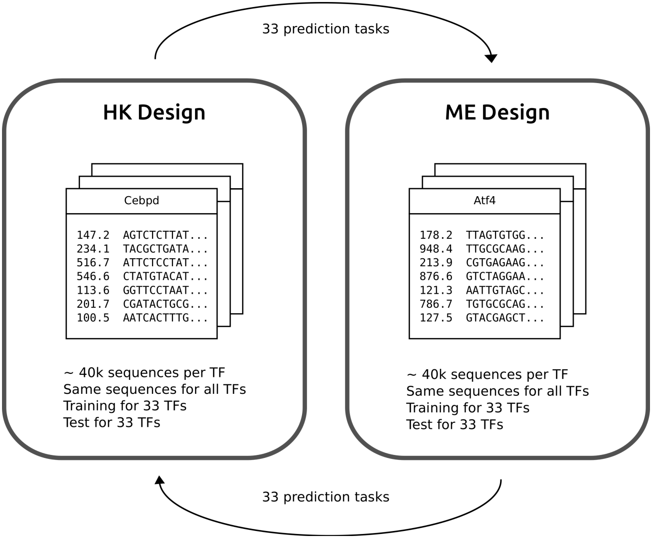
Evaluation

DREAM5 Transcription-Factor Binding Affinity Prediction
DNA-Motif Recognition Challenge in 2011
Collection of 66 regression tasks in a biological domain.

Goal Predict binding affinity of protein (TF) to a given DNA sequence

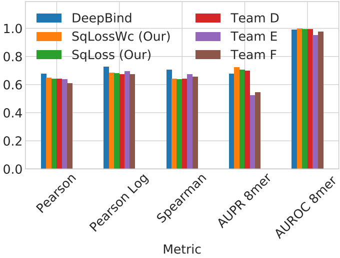
Value	Data points
290.507	AGGGCATCATGGAGCTGTCCAG
679.305	ATCACAATTTTGCCGAGAGCGA
1998.715	GTACACCCCGTTCGGCGGCCCA
447.803	CCTTTAGCCCATCGTTGGCCAA

TF Binding Affinity Prediction



TF Binding Affinity Results I

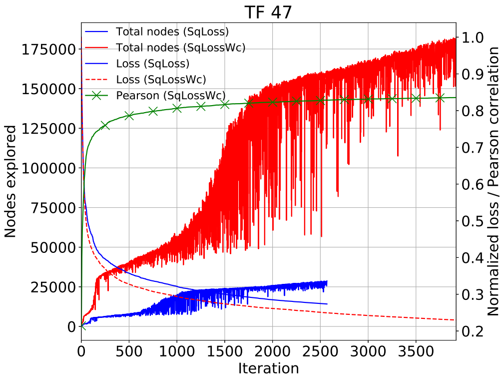
Team	Pearson	Pearson_Log	Spearman	AUPR_8mer	AUROC_8mer
DeepBind	0.6780	0.7260	0.7060	0.6760	0.9910
SqLossWC	0.6483	0.6846	0.6423	0.7236	0.9967
SqLoss	0.6399	0.6791	0.6390	0.7049	0.9953
Team_D	0.6413	0.6742	0.6394	0.6997	0.9942
Team_E	0.6375	0.6936	0.6735	0.5223	0.9524
Team_F	0.6103	0.6732	0.6555	0.5456	0.9766



- Good results across all metrics
- No domain knowledge needed
- Comparable to other linear models

TF Binding Affinity Results II

Motif	Weight
TAAT*A	0.733985
TAATG*G	0.706344
ATG*AAA	0.674507
⋮	⋮
GGATA	-0.188202
TCAAT	-0.214858
G*ATAG	-0.218132



- Final model is a list of weighted *k*-mers
- Wildcards improve results but also imply a computational burden

Conclusion & Future Work

- Linear model for sequence regression
- All-length subsequence feature space
- Mean squared error as loss function
- Coordinate descent with Gauss-Southwell selection
- Branch-and-Bound strategy for efficient GS selection

Future Work

- Expand to other data structures
- Ensemble multiple models
- Stochastic variant of our approach

Further Information

Paper, Code and Data available

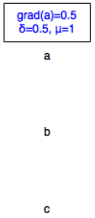
Efficient Sequence Regression by Learning Linear Models in All-Subsequence Space

Severin Gsponer, Barry Smyth, Georgiana Ifrim

Code and data available at:
github.com/svgsponer/SqLoss

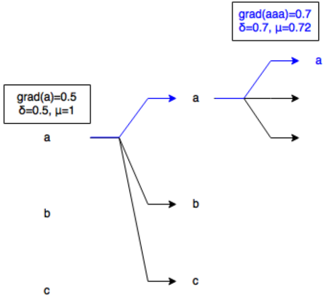
Email:
severin.gsponer@insight-centre.org

Appendix: Search and Pruning



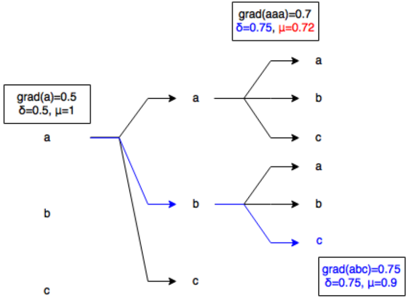
Calculate gradient and bound μ for each node. Save global maximal gradient in δ.

Appendix: Search and Pruning



Continue in a breadth first manner.

Appendix: Search and Pruning



Continue in a breadth first manner.