Naïve Discriminative Learning

From Rescorla-Wagner to Machine Learning in Natural Language Processing

Stefan Evert

Friedrich-Alexander-Universität Erlangen-Nürnberg, Germany
stefan.evert@fau.de

Osnabrück, 21 June 2017

Outline

Outline

Naïve Discriminative Learning

- Baayen (2011); Baayen *et al.* (2011)
- Incremental learning equations for direct associations between cues and outcomes (Rescorla and Wagner 1972)
- Equilibrium conditions (Danks 2003)
- Implementation as R package `ndl` (Arppe *et al.* 2014)

Naive: cue-outcome associations estimated separately for each outcome (this independence assumption is similar to a naive Bayesian classifier)

Discriminative: cues predict outcomes based on total activation level = sum of direct cue-outcome associations

Learning: incremental learning of association strengths

The Rescorla-Wagner equations (1972)

Represent incremental associative learning and subsequent on-going adjustments to an accumulating body of knowledge.

Changes in cue–outcome association strengths:

- no change if a cue is not present in the input
- increased if the cue and outcome co-occur
- decreased if the cue occurs without the outcome
- if outcome can already be predicted well (based on all cues), adjustments become smaller

Only final results of incremental adjustments to the cue–outcome associations are kept – no need for remembering the individual adjustments, however many there are.

Danks (2003) equilibrium conditions

- Presume an ideal stable "adult" state, where all cue–outcome associations have fully been learnt – further data points should then have no impact on the cue–outcome associations
- Provide a convenient short-cut to calculating the final cue–outcome association weights resulting from incremental learning, using relatively simple matrix algebra
- Most learning parameters of the Rescorla-Wagner equations drop out of the Danks equilibrium equation
- Circumvent the problem that a simulation of an R-W learner does usually not converge to a stable state unless the learning rate is gradually decreased

Traditional vs. linguistic applications of R-W

- Traditionally: simple controlled experiments on item-by-item learning, with only a handful of cues and perfect associations
- Natural language: full of choices among multiple possible alternatives – phones, words, or constructions – which are influenced by a large number of contextual factors, and which often show weak to moderate tendencies towards one or more of the alternatives rather than a single unambiguous decision
- These messy, complex types of problems are a key area of interest in modeling and understanding language use
- Application of R-W in the form of a Naïve Discriminative Learner to such linguistic classification problems is sucessful in practice and can throw new light on research questions

Examples

- Dative alternation in English
  - *Mary gave John the book* vs. *Mary gave the book to John*
  - Which factors determine the choice of construction?
- Bresnan *et al.* (2007) predict speaker choice with acc. = 92%
  - based on features such as pronominality, definiteness, number, person, animacy, concreteness, semantic class and accessibility of recipient and theme (most features have a significant effect)
  - logistic regression, baseline acc. = 79%
- Baayen (2011) obtains same result with simpler NDL
  - TiMBL: 92%, SVM: 93%, GLMM: 93%
  - NDL with speaker information: 95% (other learners don't benefit from speakers)
- Baayen *et al.* (2011) apply NDL to morphological learning based on direct form-meaning associations

Related work

- R-W vs. perceptron (Sutton and Barto 1981, p. 155f)
- R-W vs. least-squares regression (Stone 1986, p. 457)
- R-W vs. logistic regression (Gluck and Bower 1988, p. 234)
- R-W vs. neural networks (Dawson 2008)

☞ similarities are also mentioned by many other authors …

Outline

The Rescorla-Wagner equations

- Goal of naïve discriminative learner: predict an outcome O based on presence or absence of a set of cues $C_1, \dots, C_n$
- An event $(\mathbf{c}, o)$ is formally described by indicator variables

$$c_i = \begin{cases} 1 & \text{if } C_i \text{ is present} \\ 0 & \text{otherwise} \end{cases} \qquad o = \begin{cases} 1 & \text{if } O \text{ results} \\ 0 & \text{otherwise} \end{cases}$$

- Given cue-outcome associations $\mathbf{v} = (V_1, \dots, V_n)$ of learner, the activation level of the outcome O is

$$\sum_{j=1}^{n} c_j V_j$$

- Associations $\mathbf{v}^{(t)}$ as well as cue and outcome indicators $(\mathbf{c}^{(t)}, o^{(t)})$ depend on time step t

The Rescorla-Wagner equations

- Rescorla and Wagner (1972) proposed the R-W equations for the change in associations given an event $(\mathbf{c}, o)$:

$$\Delta V_i = \begin{cases} 0 & \text{if } c_i = 0 \\ \alpha_i \beta_1 \bigl(\lambda - \sum_{j=1}^{n} c_j V_j\bigr) & \text{if } c_i = 1 \wedge o = 1 \\ \alpha_i \beta_2 \bigl(0 - \sum_{j=1}^{n} c_j V_j\bigr) & \text{if } c_i = 1 \wedge o = 0 \end{cases}$$

  with parameters

$\lambda > 0$	target activation level for outcome O
$\alpha_i > 0$	salience of cue C_i
$\beta_1 > 0$	learning rate for positive events ($o = 1$)
$\beta_2 > 0$	learning rate for negative events ($o = 0$)

The Widrow-Hoff rule

- The W-H rule (Widrow and Hoff 1960) is a widely-used simplification of the R-W equations:

$$\Delta V_i = \begin{cases} 0 & \text{if } c_i = 0 \\ \beta\big(1 - \sum_{j=1}^n c_j V_j\big) & \text{if } c_i = 1 \wedge o = 1 \\ \beta\big(0 - \sum_{j=1}^n c_j V_j\big) & \text{if } c_i = 1 \wedge o = 0 \end{cases}$$
$$= c_i \beta\big(o - \sum_{j=1}^n c_j V_j\big)$$

with parameters

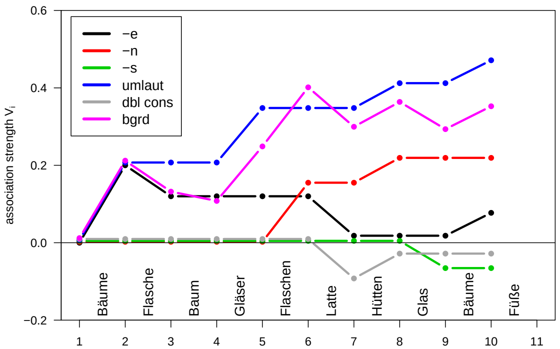
$\lambda = 1$	target activation level for outcome O
$\alpha_i = 1$	salience of cue C_i
$\beta_1 = \beta_2$ $= \beta > 0$	global learning rate for positive and negative events

A simple example: German noun plurals

t	word	o pl?	c_1 $-e$	c_2 $-n$	c_3 $-s$	c_4 umlaut	c_5 dbl cons	c_6 bgrd
1	Bäume	1	1	0	0	1	0	1
2	Flasche	0	1	0	0	0	0	1
3	Baum	0	0	0	0	0	0	1
4	Gläser	1	0	0	0	1	0	1
5	Flaschen	1	0	1	0	0	0	1
6	Latte	0	1	0	0	0	1	1
7	Hütten	1	0	1	0	1	1	1
8	Glas	0	0	0	1	0	0	1
9	Bäume	1	1	0	0	1	0	1
10	Füße	1	1	0	0	1	0	1

A simple example: German noun plurals

t	$\sum c_j V_j$	V_1	V_2	V_3	V_4	V_5	V_6
10	.882	.077	.217	-.070	.464	-.038	.340
Füße	1	1	0	0	1	0	1
	o	c_1	c_2	c_3	c_4	c_5	c_6

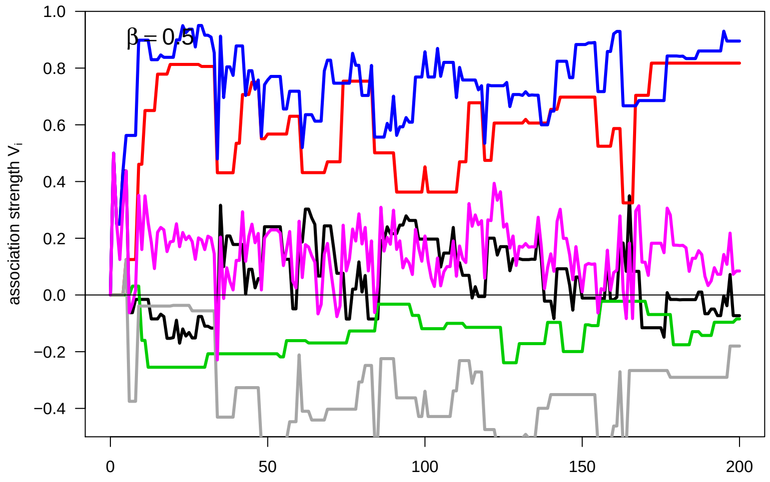


A stochastic NDL learner

- A specific event sequence $(\mathbf{c}^{(t)}, o^{(t)})$ will only be encountered in controlled experiments
- For applications in corpus linguistics, it is more plausible to assume that events are randomly sampled from a population of event tokens $(\mathbf{c}^{(k)}, o^{(k)})$ for $k = 1, \ldots, m$
  - ☞ event types listed repeatedly proportional to their frequency
- I.i.d. random variables $\mathbf{c}^{(t)} \sim \mathbf{c}$ and $o^{(t)} \sim o$
  - ☞ distributions of $\mathbf{c}$ and o determined by population
- NDL can now be trained for arbitrary number of time steps, even if population is small (as in our example)
  - study asymptotic behaviour of learners
  - convergence ➜ stable "adult" state of associations

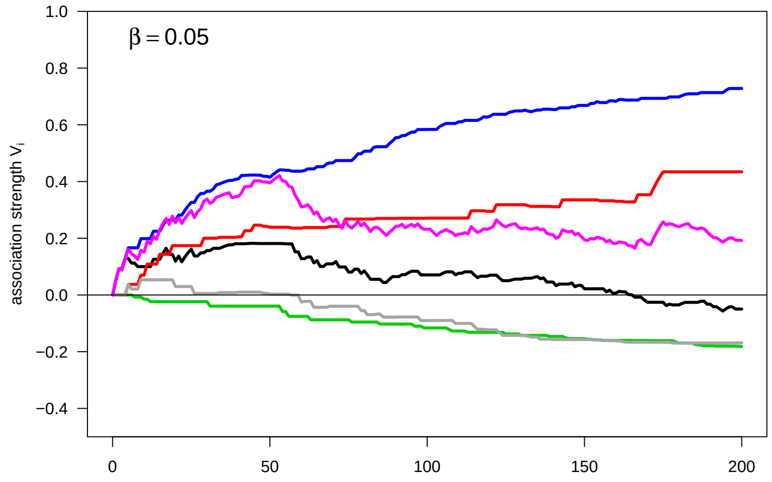
A stochastic NDL learner

Effect of the learning rate β



A stochastic NDL learner

Effect of the learning rate β



Outline

Expected activation levels

- Since we are interested in the general behaviour of a stochastic NDL, it makes sense to average over many individual learners to obtain expected associations $\mathrm{E}[V_j^{(t)}]$

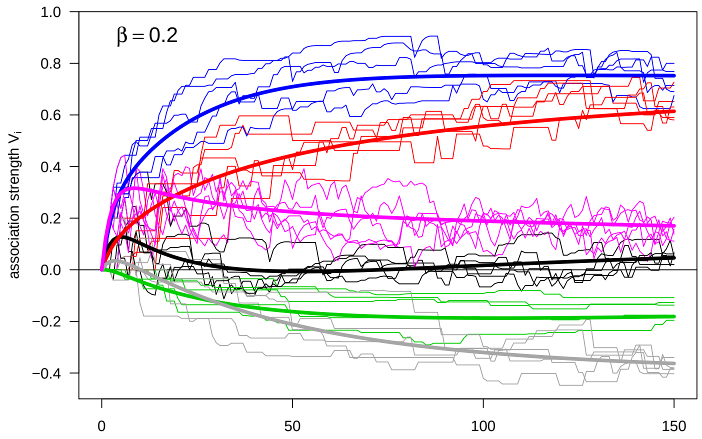
$$\mathrm{E}[V_j^{(t+1)}] = \mathrm{E}[V_j^{(t)}] + \mathrm{E}[\Delta V_j^{(t)}]$$

$$\begin{aligned}\mathrm{E}[\Delta V_j^{(t)}] &= \mathrm{E}\left[c_i \beta \big(o - \textstyle\sum_{j=1}^n c_j V_j^{(t)}\big)\right] \\ &= \beta \cdot \Big(\Pr(C_i, O) - \textstyle\sum_{j=1}^n \Pr(C_i, C_j) \mathrm{E}[V_j^{(t)}]\Big)\end{aligned}$$

- c_i and c_j are independent from $V_j^{(t)}$
- indicator variables: $\mathrm{E}[c_i o] = \Pr(C_i, O)$; $\mathrm{E}[c_i c_j] = \Pr(C_i, C_j)$

Expected activation levels

$$\mathrm{E}[\Delta V_j^{(t)}] = \beta \cdot \left(\Pr(C_i, O) - \sum_{j=1}^n \Pr(C_i, C_j)\mathrm{E}[V_j^{(t)}]\right)$$



The Danks equilibrium

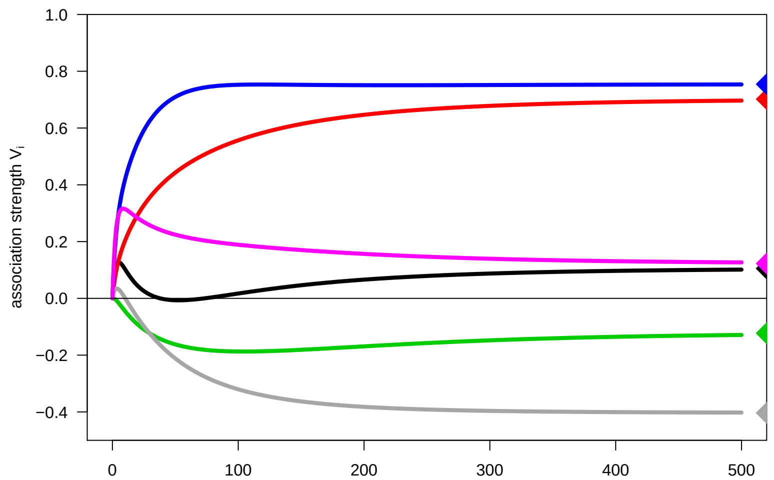
- If $\mathrm{E}[V_i^{(t)}]$ converges, the asymptote $V_i^* = \lim_{t\to\infty} \mathrm{E}[V_i^{(t)}]$ must satisfy the Danks equilibrium conditions $\mathrm{E}[\Delta V_i^*] = 0$, i.e.

$$\Pr(C_i, O) - \sum_{j=1}^n \Pr(C_i, C_j)V_j^* = 0 \quad \forall i$$

(Danks 2003, p. 113)

- Now there is a clear interpretation of the Danks equilibrium as the stable average associations reached by a community of stochastic learners with input from the same population
  - ☞ allows us to compute the "adult" state of NDL without carrying out a simulation of the learning process

The Danks equilibrium



Matrix notation

$$\mathbf{X} = \begin{bmatrix} c_1^{(1)} & \cdots & c_n^{(1)} \\ c_1^{(2)} & \cdots & c_n^{(2)} \\ \vdots & & \vdots \\ c_1^{(m)} & \cdots & c_n^{(m)} \end{bmatrix} \qquad \mathbf{z} = \begin{bmatrix} o^{(1)} \\ o^{(2)} \\ \vdots \\ o^{(m)} \end{bmatrix} \qquad \mathbf{w} = \begin{bmatrix} V^{(1)} \\ \vdots \\ V^{(n)} \end{bmatrix}$$

$$\begin{bmatrix} \Pr(C_1, O) \\ \vdots \\ \Pr(C_n, O) \end{bmatrix} = \frac{1}{m}\mathbf{X}^T\mathbf{z} \qquad \begin{bmatrix} \Pr(C_1, C_1) & \cdots & \Pr(C_1, C_n) \\ \vdots & & \vdots \\ \Pr(C_n, C_1) & \cdots & \Pr(C_n, C_n) \end{bmatrix} = \frac{1}{m}\mathbf{X}^T\mathbf{X}$$

Danks equilibrium: $\mathbf{X}^T\mathbf{z} = \mathbf{X}^T\mathbf{X}\mathbf{w}^*$

Matrix notation: German noun plurals

$$\mathbf{X} = \begin{bmatrix} 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 1 \end{bmatrix} \qquad \mathbf{z} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \\ 1 \\ 0 \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} \qquad \mathbf{w} = \begin{bmatrix} V^{(1)} \\ \vdots \\ V^{(n)} \end{bmatrix}$$

$$\begin{bmatrix} .3 \\ .2 \\ .0 \\ .5 \\ .1 \\ .6 \end{bmatrix} = \frac{1}{m}\mathbf{X}^T\mathbf{z} \qquad \begin{bmatrix} .5 & .0 & .0 & .3 & .1 & .5 \\ .0 & .2 & .0 & .1 & .1 & .2 \\ .0 & .0 & .1 & .0 & .0 & .1 \\ .3 & .1 & .0 & .5 & .1 & .5 \\ .1 & .1 & .0 & .1 & .2 & .2 \\ .5 & .2 & .1 & .5 & .2 & 1 \end{bmatrix} = \frac{1}{m}\mathbf{X}^T\mathbf{X}$$

Outline

The single-layer perceptron (SLP)

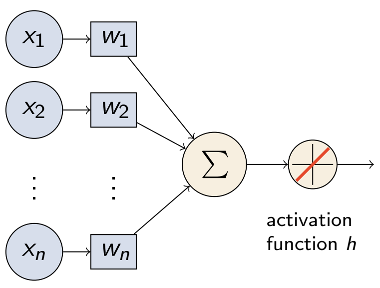
SLP (Rosenblatt 1958) is the most basic feed-forward neural network

- numeric inputs $x_1, \dots, x_n$
- output activation $h(y)$ based on weighted sum of inputs

$$y = \sum_{j=1}^{n} w_j x_j$$

- h = Heaviside step function in traditional SLP
- even simpler model: $h(y) = y$
- cost wrt. target output z:

$$E(\mathbf{w}, \mathbf{x}, z) = \left(z - \sum_{j=1}^{n} w_j x_j\right)^2$$



SLP training: the delta rule

- SLP weights are learned by gradient descent training: for a single training item $(\mathbf{x}, z)$ and learning rate $\delta > 0$

$$\begin{aligned} \Delta w_i &= -\delta \frac{\partial E(\mathbf{w}, \mathbf{x}, z)}{\partial w_i} \\ &= 2\delta x_i \left(z - \sum_{j=1}^{n} x_j w_j\right) \\ &= \beta c_i \left(o - \sum_{j=1}^{n} c_j V_j\right) \end{aligned}$$

- Perfect correspondence to W-H rule with

$$V_i = w_i \qquad c_i = x_i \qquad o = z \qquad \beta = 2\delta$$

Batch training

- Neural networks often use batch training, where all training data are considered at once instead of one item at a time
- Similar to stochastic NDL, batch training computes the expected weights $\mathrm{E}[\mathbf{w}^{(t)}]$ for a SLP with stochastic input
- The corresponding batch training cost is

$$E(\mathbf{w}) = \frac{1}{m}\sum_{k=1}^{m} E(\mathbf{w}, \mathbf{x}^{(k)}, z^{(k)}) = \frac{1}{m}\sum_{k=1}^{m} \left(z^{(k)} - \textstyle\sum_{j=1}^{n} w_j x_j^{(k)}\right)^2$$

- Minimization of $E(\mathbf{w})$ = linear least-squares regression

Linear least-squares regression

- Matrix formulation of the linear least-squares problem:

$$E(\mathbf{w}) = \frac{1}{m}\sum_{k=1}^{m} \left(z^{(k)} - \textstyle\sum_{j=1}^{n} w_j x_j^{(k)}\right)^2 = \frac{1}{m}(\mathbf{z} - \mathbf{X}\mathbf{w})^T(\mathbf{z} - \mathbf{X}\mathbf{w})$$

- Minimum of $E(\mathbf{w})$, the L_2 solution, must satisfy $\nabla E(\mathbf{w}^*) = \mathbf{0}$, which leads to the normal equations

$$\mathbf{X}^T\mathbf{z} = \mathbf{X}^T\mathbf{X}\mathbf{w}^*$$

- Normal equations = Danks equilibrium conditions
- Regression theory shows that batch training / stochastic NLP converges to the unique* solution of the L_2 problem

What have we learned?

$$\boxed{\begin{array}{ccccc} \text{stochastic} & = & \text{batch} & = & L_2 \text{ regression} \\ \text{NDL} & = & \text{SLP} & & \end{array}}$$

☞ These equivalences also hold for the general R-W equations with arbitrary values of α_i, β_1, β_2 and λ
- salience-adjusted weights: $w_i = V_i / \sqrt{\alpha_i}$
- scaled outcome indicators: $z = \lambda o$
- scaled cue indicators:

$$x_i = \begin{cases} \sqrt{\alpha_i} & \text{if } c_i = 1 \wedge o = 1 \\ \sqrt{\alpha_i}\sqrt{\frac{\beta_2}{\beta_1}} & \text{if } c_i = 1 \wedge o = 0 \\ 0 & \text{otherwise} \end{cases}$$

Effects of R-W parameters

$\beta > 0$: learning rate ➜ convergence of individual learners

$\lambda \neq 1$: linear scaling of associations / activation (obvious)

$\alpha_i \neq 1$: salience of cue C_i determines how fast associations are learned, but does not affect the final stable associations (same L_2 regression problem)

$\beta_1 \neq \beta_2$: different positive/negative learning rates *do* affect the stable associations; closely related to prevalence of positive and negative events in the population

Outline

The last 50 years of computational linguistics

1970s	Rule-based approaches	Toy worlds, expert systems
1980s	Linguistic theories	
	Formal language theory	
1990s		Large corpora
	Probabilistic models	
2000s		Supervised classification
	Machine learning	
2010s		
	Deep learning (RNN)	
2020s		End-to-end learning

NLP as supervised classification

Part-of-speech tagging

	JJ					
RB	VVP	VVP	VVZ		VVZ	
DT	NN	NN	NNS	CD	NNS	SENT
This	*light*	*brew*	*costs*	*ten*	*bucks*	*.*

Tokenization

The 14|?F model is approx|?. 15|?cm long|?.

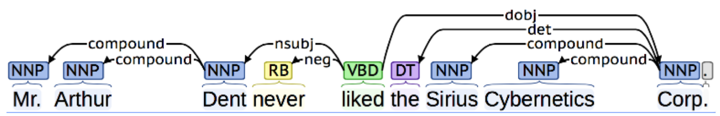
NLP as supervised classification

Syntactic attachment disambiguation

I saw the girl and the boy with glasses.

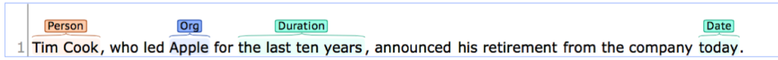
I saw [the girl and the boy] [with glasses]
I saw [[the girl and the boy] with glasses]
I saw [the girl] and [the boy with glasses]

Dependency parsing



NLP as supervised classification

Named entity recognition



Semantic role labelling
Word sense disambiguation
Sentiment analysis
Author profiling
Recognizing textual entailment
Automatic grading
. . .

Outline

The maximum entropy approach (MaxEnt)

- A popular machine learning algorithm in NLP is based on the maximum entropy principle (Berger *et al.* 1996)
- Estimate conditional distribution $p(y|x)$ with maximal entropy

$$H(p) = -\sum_{x,y} \tilde{p}(x) p(y|x) \log p(y|x)$$

so that expectations of features $f_i(x,y)$ match training data

$$\mathrm{E}_p[f_i(x,y)] = \sum_{x,y} \tilde{p}(x,y) f_i(x,y)$$

☞ These features represent associations between outcomes (y) and cues (different properties of x)

The maximum entropy approach (MaxEnt)

- Leads to a log-linear probability distribution

$$p(y|x) \sim e^{\sum_i \lambda_i f_i(x,y)}$$

  whose parameters λ_i are optimized to maximize the log-likelihood of the training data

- For binary y, this is equivalent to logistic regression
  - remember Bresnan *et al.* (2007)?

Logistic regression and NDL

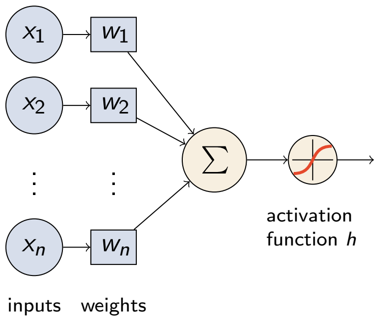
Logistic regression is the standard tool for predicting a categorical response from binary features

- can be expressed as SLP with probabilistic interpretation
- uses logistic activation function

$$h(y) = \frac{1}{1 + e^{-y}}$$

- and Bernoulli cost

$$E(\mathbf{w}, \mathbf{x}, z) = \begin{cases} -\log h(y) & \text{if } z = 1 \\ -\log(1 - h(y)) & \text{if } z = 0 \end{cases}$$



Logistic regression and NDL

- Gradient descent training leads to delta rule that corresponds to a modified version of the R-W equations

$$\Delta V_i = \begin{cases} 0 & \text{if } c_i = 0 \\ \beta \left(1 - h\left(\sum_{j=1}^{n} c_j V_j\right)\right) & \text{if } c_i = 1 \wedge o = 1 \\ \beta \left(0 - h\left(\sum_{j=1}^{n} c_j V_j\right)\right) & \text{if } c_i = 1 \wedge o = 0 \end{cases}$$

- Same as original R-W, except that activation level is now transformed into probability $h(y)$
- But no easy way to analyze stochastic learning process (batch training $\neq$ expected value of single-item training)
- Less robust for highly predictable outcomes ➜ **w** diverges

Outline

Summary & some open questions

stochastic	=	batch	=	L_2 regression
NDL	=	linear SLP		
MaxEnt	=	sigmoid SLP	=	logistic regression

- How many training steps are needed for a stochastic NDL learner to converge to the Danks equilibrium?
- What is the relation between NDL and regularized regression?
- How does logistic regression behave as incremental learner?
- Which sequences / patterns in the input data lead to significantly different behaviour from a stochastic learner?

Acknowledgements

Follow me on Twitter: @RattiTheRat

The mathematical analysis was fuelled by large amounts of coffee and cinnamon rolls at Cinnabon, Harajuku, Tokyo

Based on joint research with Antti Arppe (U Alberta). Read Evert and Arppe (2015) for the full story.

References I

Arppe, Antti; Hendrix, Peter; Milin, Petar; Baayen, R. Harald; Shaoul, Cyrus (2014). *ndl: Naive Discriminative Learning*. R package version 0.2.16.

Baayen, R. Harald (2011). Corpus linguistics and naive discriminative learning. *Brazilian Journal of Applied Linguistics*, **11**, 295–328.

Baayen, R. Harald; Milin, Petar; Đurđević, Dusica Filipović; Hendrix, Peter; Marelli, Marco (2011). An amorphous model for morphological processing in visual comprehension based on naive discriminative learning. *Psychological Review*, **118**(3), 438–81.

Berger, Adam L.; Della Pietra, Stephen A.; Della Pietra, Vincent J. (1996). A maximum entropy approach to natural language processing. *Computational Linguistics*, **22**(1), 39–71.

Bresnan, Joan; Cueni, Anna; Nikitina, Tatiana; Baayen, R. Harald (2007). Predicting the dative alternation. In G. Boume, I. Kraemer, and J. Zwarts (eds.), *Cognitive Foundations of Interpretation*, pages 69–94. Royal Netherlands Academy of Science, Amsterdam, Netherlands.

Danks, David (2003). Equilibria of the Rescorla-Wagner model. *Journal of Mathematical Psychology*, **47**, 109–121.

Dawson, Michael R. W. (2008). Connectionism and classical conditioning. *Comparative Cognition & Behavior Reviews*, **3**.

References II

Evert, Stefan and Arppe, Antti (2015). Some theoretical and experimental observations on naïve discriminative learning. In *Proceedings of the 6th Conference on Quantitative Investigations in Theoretical Linguistics (QITL-6)*, Tübingen, Germany.

Gluck, Mark A. and Bower, Gordon H. (1988). From conditioning to category learning: An adaptive network model. *Journal of Experimental Psychology: General*, **117**(3), 227–247.

Rescorla, Robert A. and Wagner, Allen R. (1972). A theory of Pavlovian conditioning: Variations in the effectiveness of reinforcement and nonreinforcement. In A. H. Black and W. F. Prokasy (eds.), *Classical Conditioning II: Current Research and Theory*, chapter 3, pages 64–99. Appleton-Century-Crofts, New York.

Rosenblatt, Frank (1958). The perceptron: A probabilistic model for information storage and organization in the brain. *Psychological Review*, **65**(6), 386–408.

Stone, G. O. (1986). An analysis of the delta rule and the learning of statistical associations. In D. E. Rumelhart and J. L. McClelland (eds.), *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations*, chapter 11, pages 444–459. MIT Press, Cambridge, MA.

Sutton, Richard S. and Barto, Andrew G. (1981). Toward a modern theory of adaptive networks: Expectation and prediction. *Psychological Review*, **88**(2), 135–170.

References III

Widrow, Bernard and Hoff, Marcian E. (1960). Adaptive switching circuits. In *IRE WESCON Convention Record*, pages 96–104, New York. IRE.