

Maxent and Classification

CMSC 473/673

Frank Ferraro

Outline

Maximum Entropy models

Defining the model

Defining the objective

Learning: Optimizing the objective

Math: gradient derivation (advanced)

Terminology

common NLP
term

Log-Linear Models

as statistical
regression

(Multinomial) logistic regression

Softmax regression

based in
information theory

Maximum Entropy models (MaxEnt)

a form of

Generalized Linear Models

viewed as

Discriminative Naïve Bayes

to be cool
today :)

Very shallow (sigmoidal) neural nets

Maxent Models are Flexible

Maxent models can be used:

- to design discriminatively trained classifiers, or
- to create featureful language models

Maxent Models for Classification: Discriminatively ...

*Directly model
the posterior*

$$p(Y | X) = \textbf{maxent}(X; Y)$$

Discriminatively trained classifier

Maxent Models for Classification: Discriminatively or Generatively Trained

*Directly model
the posterior*

$$p(Y | X) = \text{maxent}(X; Y)$$

Discriminatively trained classifier

*Model the
posterior with
Bayes rule*

$$p(Y | X) \propto \text{maxent}(X | Y)p(Y)$$

*Generatively trained classifier with
maxent-based language model*

Maximum Entropy (Log-linear) Models For Discriminatively Trained Classifiers

(we'll start with this one)

$$p(y \mid x) = \text{maxent}(x, y)$$

*discriminatively trained:
classify in one go*

Discriminative ML Classification in 30 Seconds

- Common goal: probabilistic classifier $p(y \mid x)$
- Often done by defining **features** between x and y that are meaningful

Discriminative ML Classification in 30 Seconds

- Common goal: probabilistic classifier $p(y \mid x)$
- Often done by defining **features** between x and y that are meaningful
 - Denoted by a general vector of K features

$$f(x, y) = (f_1(x, y), \dots, f_K(x, y))$$

Discriminative ML Classification in 30 Seconds

- Common goal: probabilistic classifier $p(y \mid x)$
- Often done by defining **features** between x and y that are meaningful
 - Denoted by a general vector of K features
$$f(x, y) = (f_1(x, y), \dots, f_K(x, y))$$
- Features can be thought of as “soft” rules
 - E.g., POSITIVE sentiments tweets *may* be more likely to have the word “happy”

Core Aspects to Maxent Classifier

$p(y|x)$

We need to define

- **features** $f(x, y)$ between x and y that are meaningful;
- **weights** θ (one per feature) to say how important each feature is; and
- a way to **form probabilities** from f and θ

Reminder!

The NLP Research Community: Standard Tasks (some of them)

GLUE

<https://gluebenchmark.com/>

GLUE Tasks

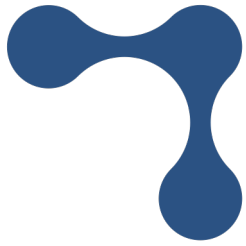
Name	Download
The Corpus of Linguistic Acceptability	
The Stanford Sentiment Treebank	
Microsoft Research Paraphrase Corpus	
Semantic Textual Similarity Benchmark	
Quora Question Pairs	
MultiNLI Matched	
MultiNLI Mismatched	
Question NLI	
Recognizing Textual Entailment	
Winograd NLI	
Diagnostics Main	

SuperGLUE Tasks

Name	Identifier
Broadcoverage Diagnostics	AX-b
CommitmentBank	CB
Choice of Plausible Alternatives	COPA
Multi-Sentence Reading Comprehension	MultiRC
Recognizing Textual Entailment	RTE
Words in Context	WiC
The Winograd Schema Challenge	WSC
BoolQ	BoolQ
Reading Comprehension with Commonsense Reasoning	ReCoRD
Winogender Schema Diagnostics	AX-g

 **SuperGLUE**

<https://super.gluebenchmark.com/>

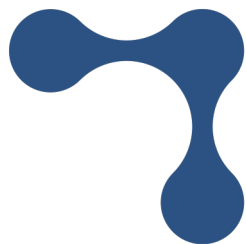


Recognizing Textual Entailment (RTE)

Given a premise sentence s and hypothesis sentence h , determine if h “follows from” s

ENTAILMENT (yes):

NOT ENTAILED (no):



Recognizing Textual Entailment (RTE)

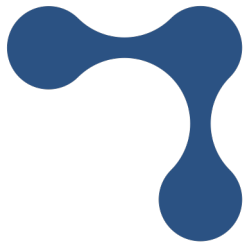
Given a premise sentence s and hypothesis sentence h , determine if h “follows from” s

ENTAILMENT (yes):

s : Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the Chicago Bulls to six National Basketball Association championships.

h : The Bulls basketball team is based in Chicago.

NOT ENTAILED (no):



Recognizing Textual Entailment (RTE)

Given a premise sentence **s** and hypothesis sentence **h**,
determine if **h** “follows from” **s**

ENTAILMENT (yes):

s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the Chicago Bulls to six National Basketball Association championships.

h: The Bulls basketball team is based in Chicago.

NOT ENTAILED (no):

s: Based on a worldwide study of smoking-related fire and disaster data, UC Davis epidemiologists show smoking is a leading cause of fires and death from fires globally.

h: Domestic fires are the major cause of fire death.

RTE

s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the Chicago Bulls to six National Basketball Association championships.

h: The Bulls basketball team is based in Chicago.

ENTAILED

p(

ENTAILED

|

s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the Chicago Bulls to six National Basketball Association championships.
h: The Bulls basketball team is based in Chicago.

)

Discriminative Document Classification

s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the Chicago Bulls to six National Basketball Association championships.

ENTAILED

h: The Bulls basketball team is based in Chicago.

Discriminative Document Classification

s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the **Chicago** Bulls to six National Basketball Association championships.

h: The Bulls basketball team is based in **Chicago**.

ENTAILED

These extractions are all **features** that have **fired** (likely have some significance)

Discriminative Document Classification

s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the **Chicago** Bulls to six National Basketball Association championships.

h: The Bulls basketball team is based in **Chicago**.

ENTAILED

These extractions are all **features** that have **fired** (likely have some significance)

Discriminative Document Classification

s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the **Chicago** Bulls to six National **Basketball** Association championships.

h: The Bulls **basketball** team is based in **Chicago**.

ENTAILED

These extractions are all **features** that have **fired** (likely have some significance)

We need to *score* the
different extracted clues.

extract_and_score_{Bulls, entailed}(, ENTAILED)

ENTAILED

including Scottie Pippen, took
the **Chicago** Bulls to six
National **Basketball**
Association championships.

extract_and_score_{basketball, entailed}(, ENTAILED)

h: The Bulls **basketball** team
is based in **Chicago**.

extract_and_score_{Chicago, entailed}(, ENTAILED)

Score and Combine Our Clues

$\text{score}_1(\text{document}, \text{ENTAILED})$

$\text{score}_2(\text{document}, \text{ENTAILED})$

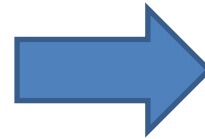
$\text{score}_3(\text{document}, \text{ENTAILED})$

...

$\text{score}_k(\text{document}, \text{ENTAILED})$

...

COMBINE



posterior
probability of
ENTAILED

Scoring Our Clues

score(

s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the Chicago Bulls to six National Basketball Association championships.

h: The Bulls basketball team is based in Chicago.

, ENTAILED) =

*(ignore the
feature indexing
for now)*

score₁(, ENTAILED)

+

score₂(, ENTAILED)

+

score₃(, ENTAILED)

+

...

Scoring Our Clues

score(

s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the Chicago Bulls to six National Basketball Association championships.

h: The Bulls basketball team is based in Chicago.

, ENTAILED) =

Learn these scores... but how?

What do we optimize?

score₁(, ENTAILED)

+

score₂(, ENTAILED)

+

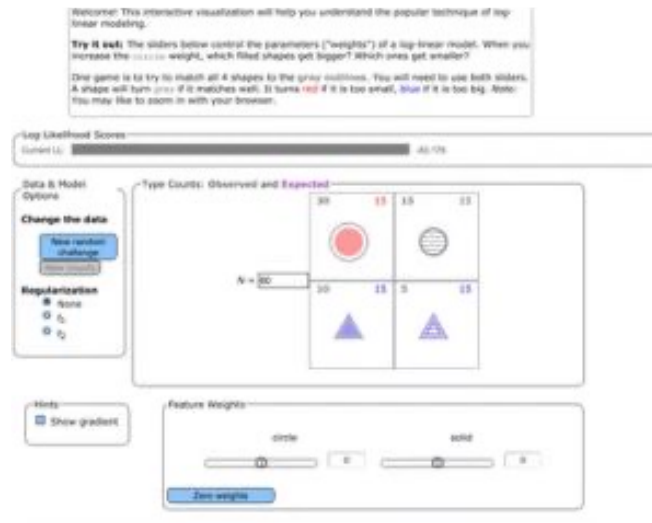
score₃(, ENTAILED)

+

...

<https://www.csee.umbc.edu/courses/undergraduate/473/f21/loglin-tutorial/>

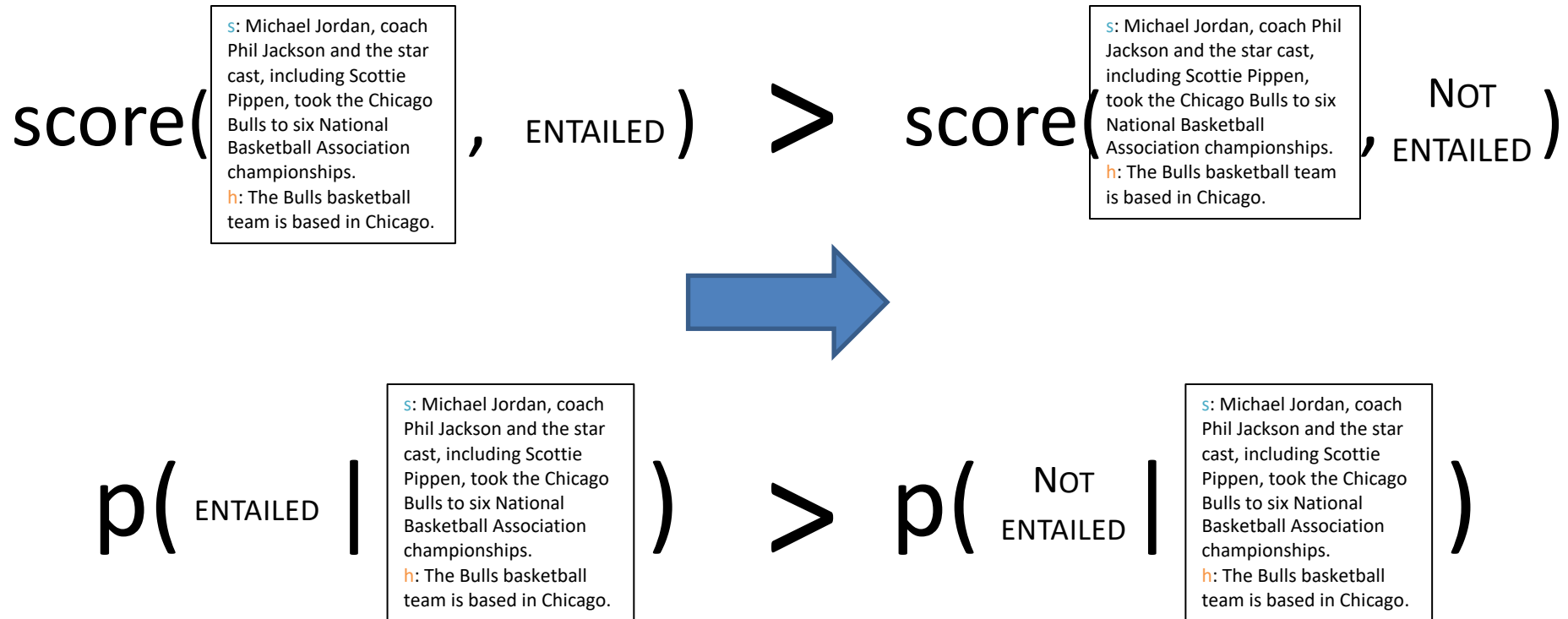
Lesson 1



Authors: Frank Fariello and Jason Eisner.
Find a bug? Want a feature? Think something can be improved? Please fill out an [issue report](#).
Last change:

The source code is released [here](#) under the [GNU Affero 3.0 license](#).
The [visual tool](#) and materials are licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](#).
[CC-BY-SA](#)
To refer to this work, cite the paper that describes it [\(Fariello & Eisner, 2013\)](#).

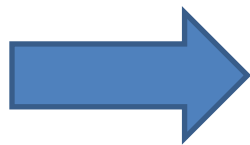
Turning Scores into Probabilities



KEY IDEA

Turning Scores into Probabilities (More Generally)

$$\text{score}(x, y_1) > \text{score}(x, y_2)$$



$$p(y_1 | x) > p(y_2 | x)$$

KEY IDEA

Maxent Modeling

$$p(\text{ENTAILED} \mid \text{s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the Chicago Bulls to six National Basketball Association championships. h: The Bulls basketball team is based in Chicago.}) \propto$$

Convert through
function G ?

What is this
function?

$$G(\text{score}(\text{s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the Chicago Bulls to six National Basketball Association championships. h: The Bulls basketball team is based in Chicago.}, \text{ENTAILED}))$$

This could be any real number

What function G...

operates on any real number?

is never less than 0?

monotonic? ($a < b \rightarrow G(a) < G(b)$)

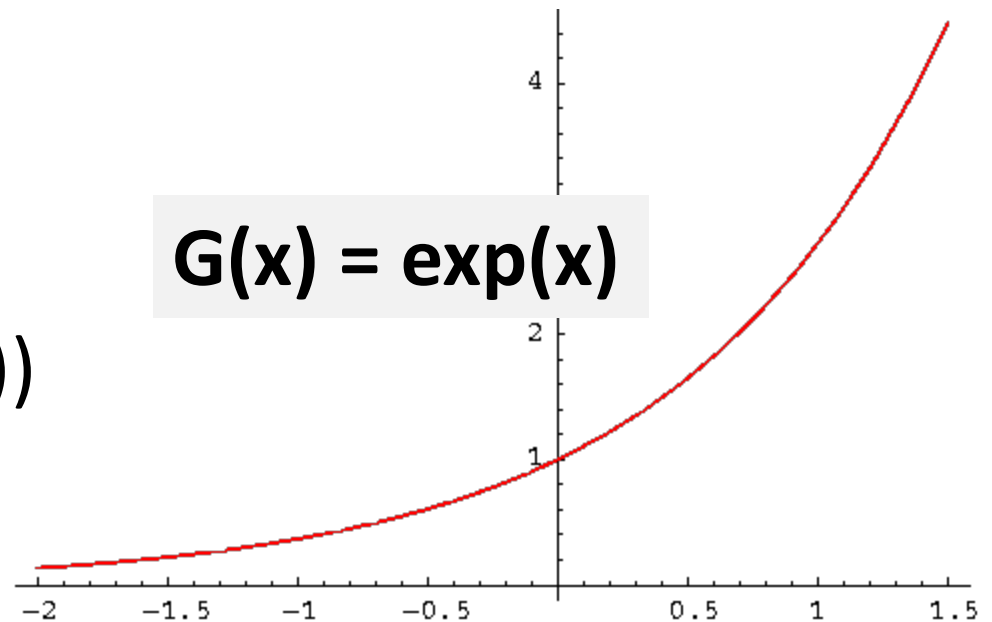
What function G...

operates on any real number?

is never less than 0?

monotonic?

$(a < b \rightarrow G(a) < G(b))$



$$G(x) = \exp(x)$$

Maxent Modeling

$p(\text{ENTAILED} \mid$

s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the Chicago Bulls to six National Basketball Association championships.

h: The Bulls basketball team is based in Chicago.

$\exp(\text{score}($

s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the Chicago Bulls to six National Basketball Association championships.
h: The Bulls basketball team is based in Chicago.

$, \text{ENTAILED}))$

Maxent Modeling

$$p(\text{ENTAILED} \mid \begin{array}{l} \text{s: Michael Jordan, coach Phil Jackson and the star cast,} \\ \text{including Scottie Pippen, took the Chicago Bulls to six} \\ \text{National Basketball Association championships.} \\ \text{h: The Bulls basketball team is based in Chicago.} \end{array}) \propto \exp(\begin{array}{l} \text{score}_1(\text{📄}, \text{ENTAILED}) \\ \text{score}_2(\text{📄}, \text{ENTAILED}) \\ \text{score}_3(\text{📄}, \text{ENTAILED}) \\ \dots \end{array} + \dots))$$

Maxent Modeling

$$p(\text{ENTAILED} \mid \begin{array}{l} \text{s: Michael Jordan, coach Phil Jackson and the star cast,} \\ \text{including Scottie Pippen, took the Chicago Bulls to six} \\ \text{National Basketball Association championships.} \\ \text{h: The Bulls basketball team is based in Chicago.} \end{array}) \propto \exp(\begin{array}{l} \text{score}_1(\text{📄}, \text{ENTAILED}) \\ \text{score}_2(\text{📄}, \text{ENTAILED}) \\ \text{score}_3(\text{📄}, \text{ENTAILED}) \\ \dots \end{array} +))$$

Learn the scores (but we'll declare what combinations should be looked at)

Maxent Modeling

$p(\text{ENTAILED} \mid$

s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the Chicago Bulls to six National Basketball Association championships.

h: The Bulls basketball team is based in Chicago.

$) \propto$

$\exp($

$\text{weight}_1 * \text{applies}_1(\text{📄}, \text{ENTAILED})$

$\text{weight}_2 * \text{applies}_2(\text{📄}, \text{ENTAILED})$

$\text{weight}_3 * \text{applies}_3(\text{📄}, \text{ENTAILED})$

...

$+ + +))$

Maxent Modeling

$p(\text{ENTAILED} \mid$

s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the Chicago Bulls to six National Basketball Association championships.

h: The Bulls basketball team is based in Chicago.

$) \propto$

$\exp(\text{weight}_1 * \text{applies}_1(\text{document}, \text{ENTAILED})$
 $\text{weight}_2 * \text{applies}_2(\text{document}, \text{ENTAILED})$
 $\text{weight}_3 * \text{applies}_3(\text{document}, \text{ENTAILED})$

$+ \dots +))$

K different **weights**... for K different **features**

Maxent Modeling

$p(\text{ENTAILED} \mid$

s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the Chicago Bulls to six National Basketball Association championships.

h: The Bulls basketball team is based in Chicago.

$) \propto$

$\exp(\text{weight}_1 * \text{applies}_1(\text{document}, \text{ENTAILED})$
 $\text{weight}_2 * \text{applies}_2(\text{document}, \text{ENTAILED})$
 $\text{weight}_3 * \text{applies}_3(\text{document}, \text{ENTAILED})$
 $+ \dots +)$

K different
weights...

for K different
features...

multiplied and
then summed

Maxent Modeling

$p(\text{ENTAILED} \mid \text{[document]}) \propto$

s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the Chicago Bulls to six National Basketball Association championships.

h: The Bulls basketball team is based in Chicago.

$\exp(\text{Dot_product of } \text{weight_vec} \text{ feature_vec}(\text{[document]}, \text{ENTAILED}))$

K different
weights...

for K different
features...

multiplied and
then **summed**

Maxent Modeling

$p(\text{ENTAILED} \mid \text{ }) \propto$

s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the Chicago Bulls to six National Basketball Association championships.

h: The Bulls basketball team is based in Chicago.

$\exp(\theta^T f(\text{document}, \text{ENTAILED}))$

K different
weights...

for K different
features...

multiplied and
then summed

Maxent Modeling

s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the Chicago Bulls to six National Basketball Association championships.

h: The Bulls basketball team is based in Chicago.

Q: How do we define Z?

Normalization for Classification

$$Z =$$

$$\sum_{\text{label } y} \exp(\theta^T f(\text{document icon}, y))$$

$$p(y | x) \propto \exp(\theta^T f(x, y))$$

classify doc x with label y in one go

Normalization for Classification (long form)

$$Z =$$

$$\sum_{\text{label } y} \exp(\text{weight}_1 * \text{applies}_1(\text{📄}, y) + \text{weight}_2 * \text{applies}_2(\text{📄}, y) + \text{weight}_3 * \text{applies}_3(\text{📄}, y) + \dots)$$

$$p(y \mid x) \propto \exp(\theta^T f(x, y))$$

classify doc x with label y in one go

Core Aspects to Maxent Classifier

$p(y|x)$

- **features** $f(x, y)$ between x and y that are meaningful;
- **weights** θ (one per feature) to say how important each feature is; and
- a way to **form probabilities** from f and θ

$$p(y|x) = \frac{\exp(\theta^T f(x, y))}{\sum_{y'} \exp(\theta^T f(x, y'))}$$

Different Notation, Same Meaning

$$p(y | x) = \frac{\exp(\theta^T f(x, y))}{\sum_{y'} \exp(\theta^T f(x, y'))}$$

$$p(y | x) \propto \exp(\theta^T f(x, y))$$

$$p(y | x) = \text{softmax}(\theta^T f(x, y))$$

Outline

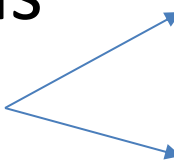
Maximum Entropy models

Defining the model

Defining the objective

Learning: Optimizing the objective

Math: gradient derivation (advanced)

- 
1. Defining Appropriate Features
 2. Understanding features in conditional models

Defining Appropriate Features in a Maxent Model

Feature functions help extract useful features (characteristics) of the data

They turn *data* into *numbers*

Features that are not 0 are said to have fired

Generally *templated*

Often binary-valued (0 or 1), but can be real-valued



Reminder!

Bag-of-words as a Function

Based on *some* tokenization, turn an input document into an array (or dictionary or set) of its unique vocab items

Think of getting a BOW rep. as a function f

input: Document

output: Container of size E , indexable by
each vocab type v

Some Bag-of-words Functions

Reminder!

Kind	Type of f_v	Interpretation
Binary	0, 1	Did v appear in the document?
Count-based	Natural number (int ≥ 0)	How often did v occur in the document?
Averaged	Real number (≥ 0 , ≤ 1)	How often did v occur in the document, normalized by doc length?
TF-IDF (term frequency, inverse document frequency)	Real number (≥ 0)	How frequent is a word, tempered by how prevalent it is across the corpus (to be covered later!)
...		

Q: Is this a reasonable representation?

Q: What are some tradeoffs (benefits vs. costs)?

Templated Features

Define a feature $f_{\text{clue}}(\text{document}, \text{label})$ for each clue you want to consider

The feature f_{clue} fires if the clue applies to/can be found in the $(\text{document}, \text{label})$ pair

Clue is often a target phrase (an n-gram) and a label

Templated Features

Define a feature $f_{\text{clue}}(\text{document}, \text{label})$ for each clue you want to consider

The feature f_{clue} fires if the clue applies to/can be found in the $(\text{document}, \text{label})$ pair

Clue is often a target phrase (an n-gram) and a label

Q: For a classifier $p(\text{label} \mid \text{document})$
are clues that depend only on
document useful?

Maxent Modeling: Templated Binary Feature Functions

$$p(\text{ENTAILED} \mid \text{document}) \propto \exp\left(\text{weight}_1 * \text{applies}_1(\text{document}, \text{ENTAILED}) + \text{weight}_2 * \text{applies}_2(\text{document}, \text{ENTAILED}) + \text{weight}_3 * \text{applies}_3(\text{document}, \text{ENTAILED}) + \dots\right)$$

s: Michael Jordan, coach Phil Jackson and the star cast, including Scottie Pippen, took the Chicago Bulls to six National Basketball Association championships.

h: The Bulls basketball team is based in Chicago.

$\text{applies}_{\text{target}, \text{type}}(\text{document}, \text{ENTAILED}) =$

$$\begin{cases} 1, & \text{target matches document and type == ENTAILED} \\ 0, & \text{otherwise} \end{cases}$$

binary

Example of a Templated Binary Feature Functions

$$\text{applies}_{\text{target,type}}(\text{Ⓜ}, \text{ENTAILED}) = \begin{cases} 1, & \text{target matches } \text{Ⓜ} \text{ and type} == \text{ENTAILED} \\ 0, & \text{otherwise} \end{cases}$$



$$\text{applies}_{\text{ball,ENTAILED}}(\text{Ⓜ}, \text{ENTAILED}) = \begin{cases} 1, & \text{ball in both s and h of } \text{Ⓜ} \text{ and ENTAILED} == \text{ENTAILED} \\ 0, & \text{otherwise} \end{cases}$$

Example of a Templated Binary Feature Functions

$$\text{applies}_{\text{target,type}}(\text{[]}, \text{ENTAILED}) = \begin{cases} 1, & \text{target matches [] and type == ENTAILED} \\ 0, & \text{otherwise} \end{cases}$$



$$\text{applies}_{\text{ball,ENTAILED}}(\text{[]}, \text{ENTAILED}) = \begin{cases} 1, & \text{ball in both s and h of [] and ENTAILED == ENTAILED} \\ 0, & \text{otherwise} \end{cases}$$

Q: What does this function check?

Example of a Templated Binary Feature Functions

$$\text{applies}_{\text{target,type}}(\text{Ⓜ}, \text{ENTAILED}) = \begin{cases} 1, & \text{target matches } \text{Ⓜ} \text{ and type} == \text{ENTAILED} \\ 0, & \text{otherwise} \end{cases}$$

$$\text{applies}_{\text{ball,ENTAILED}}(\text{Ⓜ}, \text{ENTAILED}) = \begin{cases} 1, & \text{ball in both s and h of } \text{Ⓜ} \text{ and} \\ & \text{ENTAILED} == \text{ENTAILED} \\ 0, & \text{otherwise} \end{cases}$$

Q: If there are V vocab types and L label types:

1. How many features are defined if unigram targets are used (w/ each label)?

Example of a Templated Binary Feature Functions

$$\text{applies}_{\text{target,type}}(\text{doc}, \text{ENTAILED}) = \begin{cases} 1, & \text{target matches doc and type == ENTAILED} \\ 0, & \text{otherwise} \end{cases}$$

$$\text{applies}_{\text{ball,ENTAILED}}(\text{doc}, \text{ENTAILED}) = \begin{cases} 1, & \text{ball in both s and h of doc and} \\ & \text{ENTAILED == ENTAILED} \\ 0, & \text{otherwise} \end{cases}$$

Q: If there are V vocab types and L label types:

1. How many features are defined if unigram targets are used (w/ each label)?

A1: VL

Example of a Templated Binary Feature Functions

$$\text{applies}_{\text{target,type}}(\text{Ⓢ}, \text{ENTAILED}) = \begin{cases} 1, & \text{target matches } \text{Ⓢ} \text{ and type} == \text{ENTAILED} \\ 0, & \text{otherwise} \end{cases}$$

$$\text{applies}_{\text{ball,ENTAILED}}(\text{Ⓢ}, \text{ENTAILED}) = \begin{cases} 1, & \text{ball in both s and h of } \text{Ⓢ} \text{ and} \\ & \text{ENTAILED} == \text{ENTAILED} \\ 0, & \text{otherwise} \end{cases}$$

Q: If there are V vocab types and L label types:

1. How many features are defined if unigram targets are used (w/ each label)?

A1: VL

2. How many features are defined if bigram targets are used?

Example of a Templated Binary Feature Functions

$$\text{applies}_{\text{target,type}}(\text{Ⓜ}, \text{ENTAILED}) = \begin{cases} 1, & \text{target matches } \text{Ⓜ} \text{ and type} == \text{ENTAILED} \\ 0, & \text{otherwise} \end{cases}$$

$$\text{applies}_{\text{ball,ENTAILED}}(\text{Ⓜ}, \text{ENTAILED}) = \begin{cases} 1, & \text{ball in both s and h of } \text{Ⓜ} \text{ and} \\ & \text{ENTAILED} == \text{ENTAILED} \\ 0, & \text{otherwise} \end{cases}$$

Q: If there are V vocab types and L label types:

1. How many features are defined if unigram targets are used (w/ each label)?

A1: VL

2. How many features are defined if bigram targets are used (w/ each label)?

A2: V^2L

Example of a Templated Binary Feature Functions

$$\text{applies}_{\text{target,type}}(\text{Ⓢ}, \text{ENTAILED}) = \begin{cases} 1, & \text{target matches } \text{Ⓢ} \text{ and type} == \text{ENTAILED} \\ 0, & \text{otherwise} \end{cases}$$

$$\text{applies}_{\text{ball,ENTAILED}}(\text{Ⓢ}, \text{ENTAILED}) = \begin{cases} 1, & \text{ball in both s and h of } \text{Ⓢ} \text{ and} \\ & \text{ENTAILED} == \text{ENTAILED} \\ 0, & \text{otherwise} \end{cases}$$

Q: If there are V vocab types and L label types:

1. How many features are defined if unigram targets are used (w/ each label)?

A1: VL

2. How many features are defined if bigram targets are used (w/ each label)?

A2: V^2L

3. How many features are defined if unigram and bigram targets are used (w/ each label)?

Example of a Templated Binary Feature Functions

$$\text{applies}_{\text{target,type}}(\text{Ⓜ}, \text{ENTAILED}) = \begin{cases} 1, & \text{target matches } \text{Ⓜ} \text{ and type} == \text{ENTAILED} \\ 0, & \text{otherwise} \end{cases}$$

$$\text{applies}_{\text{ball,ENTAILED}}(\text{Ⓜ}, \text{ENTAILED}) = \begin{cases} 1, & \text{ball in both s and h of } \text{Ⓜ} \text{ and} \\ & \text{ENTAILED} == \text{ENTAILED} \\ 0, & \text{otherwise} \end{cases}$$

Q: If there are V vocab types and L label types:

1. How many features are defined if unigram targets are used (w/ each label)?

A1: VL

2. How many features are defined if bigram targets are used (w/ each label)?

A2: V^2L

3. How many features are defined if unigram and bigram targets are used (w/ each label)?

A2: $(V + V^2)L$

Understanding Conditioning

$$p(y \mid x) \propto \text{count}(x)$$

Q: Is this a good model?

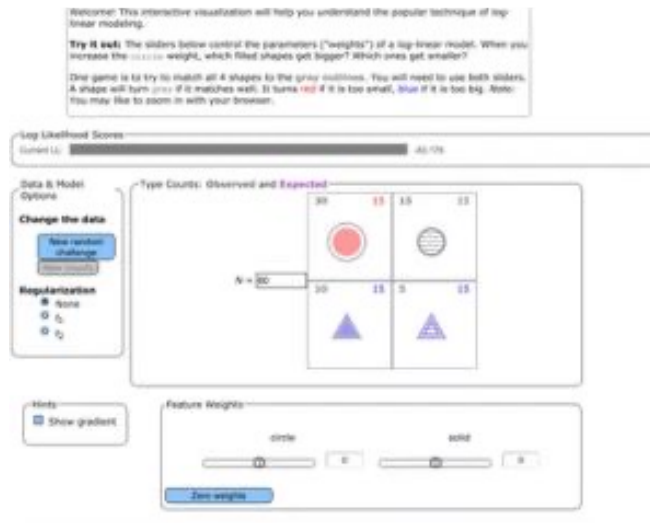
Understanding Conditioning

$$p(y \mid x) \propto \exp(\theta \cdot f(x))$$

Q: Is this a good model?

<https://www.csee.umbc.edu/courses/undergraduate/473/f21/loglin-tutorial/>

Lesson 11



Authors: Frank Fariello and Jason Eisner.
Find a bug? Want a feature? Think something can be improved? Please file out an [issue report](#).
Last change:

The source code is released [here](#) under the [GNU Affero 3.0 license](#).
The [visual tool](#) and materials are licensed under a [Creative Commons Attribution-ShareAlike 3.0 Unported License](#).
[CC-BY-SA](#)
To refer to this work, cite the paper that describes it [\(Fariello & Eisner, 2013\)](#).

Quick Comment on Features in Common Toolkits

Many toolkits have you only define features as $f(\text{document})$, and not $f(\text{document}, y)$

E.g., tensorflow, pytorch, **sklearn**, etc.

https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

```
>>> from sklearn.datasets import load_iris
>>> from sklearn.linear_model import LogisticRegression
>>> X, y = load_iris(return_X_y=True)
```

features on
data

labels

```
>>> clf = LogisticRegression(random_state=0).fit(X, y)
```

... so why is this okay?!

Outline

Maximum Entropy models

Defining the model

Defining the objective

Learning: Optimizing the objective

Math: gradient derivation (advanced)

$p_{\theta}(u \mid v)$ probabilistic model



$F(\theta; u, v)$ objective

Primary Objective: Likelihood

- Goal: maximize the score your model gives to the training data it observes
- This is called the **likelihood of your data**
- In classification, this is $p(\text{label} \mid \text{document})$
- For language modeling, this is $p(\text{word} \mid \text{label})$

Objective = Full Likelihood? (Classification)

$$\prod_i p_{\theta}(y_i|x_i) \propto \prod_i \exp(\theta^T f(x_i, y_i))$$

These values can have very small magnitude → underflow

Differentiating this product could be a pain

Logarithms

$$(0, 1] \rightarrow (-\infty, 0]$$

Products $\rightarrow$ Sums

$$\log(ab) = \log(a) + \log(b)$$

$$\log(a/b) = \log(a) - \log(b)$$

Inverse of exp

$$\log(\exp(x)) = x$$

Log-Likelihood (Classification)

Wide range of (negative) numbers

Sums are more stable

$$\log \prod_i p_{\theta}(y_i | x_i) = \sum_i \log p_{\theta}(y_i | x_i)$$

Products → Sums

$$\log(ab) = \log(a) + \log(b)$$

$$\log(a/b) = \log(a) - \log(b)$$

Log-Likelihood (Classification)

Wide range of (negative) numbers

Sums are more stable

$$\log \prod_i p_{\theta}(y_i | x_i) = \sum_i \log p_{\theta}(y_i | x_i)$$

Inverse of exp
 $\log(\exp(x)) = x$

$$= \sum_i \underbrace{\theta^T f(x_i, y_i) - \log Z(x_i)}$$

Differentiating this
becomes nicer (even
though Z depends on θ)

Log-Likelihood (Classification)

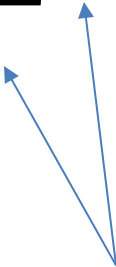
Wide range of (negative) numbers

Sums are more stable

$$\begin{aligned}\log \prod_i p_{\theta}(y_i | x_i) &= \sum_i \log p_{\theta}(y_i | x_i) \\ &= \sum_i \theta^T f(x_i, y_i) - \log Z(x_i) \\ &= F(\theta)\end{aligned}$$

Preventing Extreme Values

- Likelihood on its own can lead to overfitting and/or extreme values in the probability computation

$$F(\theta) = \sum_i \theta^T f(x_i, y_i) - \log Z(x_i)$$


Learn the parameters based on
some (fixed) data/examples

Regularization: Preventing Extreme Values

$$F(\theta) = \sum_i \theta^T f(x_i, y_i) - \log Z(x_i)$$


With fixed/predefined features, the values of θ determine how “good” or “bad” our objective learning is

Regularization: Preventing Extreme Values

$$F(\theta) = \left(\sum_i \theta^T f(x_i, y_i) - \log Z(x_i) \right) - R(\theta)$$

With fixed/predefined features, the values of θ determine how “good” or “bad” our objective learning is

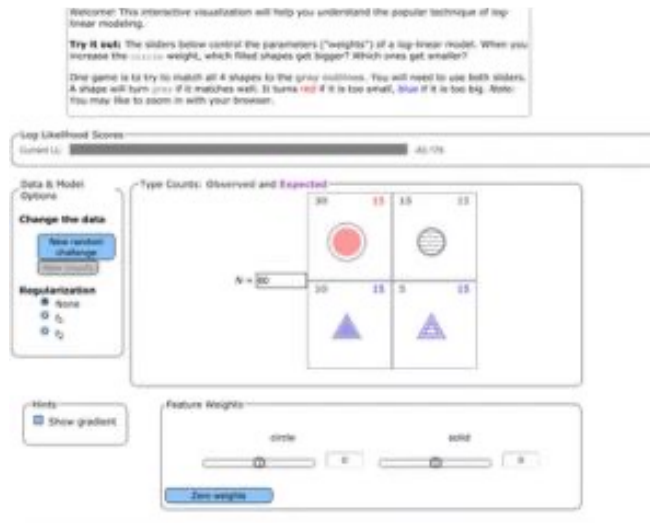
- Augment the objective with a **regularizer**
- This regularizer places an inductive bias (or, prior) on the general “shape” and values of θ

(Squared) L2 Regularization

$$R(\theta) = \|\theta\|_2^2 = \sum_k \theta_k^2$$

<https://www.csee.umbc.edu/courses/undergraduate/473/f21/loglin-tutorial/>

Lesson 8



Authors: Frank Fariello and Jason Eisner.
Find a bug? Want a feature? Think something can be improved? Please fill out an [issue report](#).
Last change:

The source code is released [here](#) under the [GNU Affero 3.0 license](#).
The [visual tool](#) and materials are licensed under a [Creative Commons Attribution-ShareAlike 4.0 International license](#).
[CC BY-SA](#)
To refer to this work, cite the paper that describes it [\(Fariello & Eisner, 2013\)](#).

Outline

Maximum Entropy classifiers

Defining the model

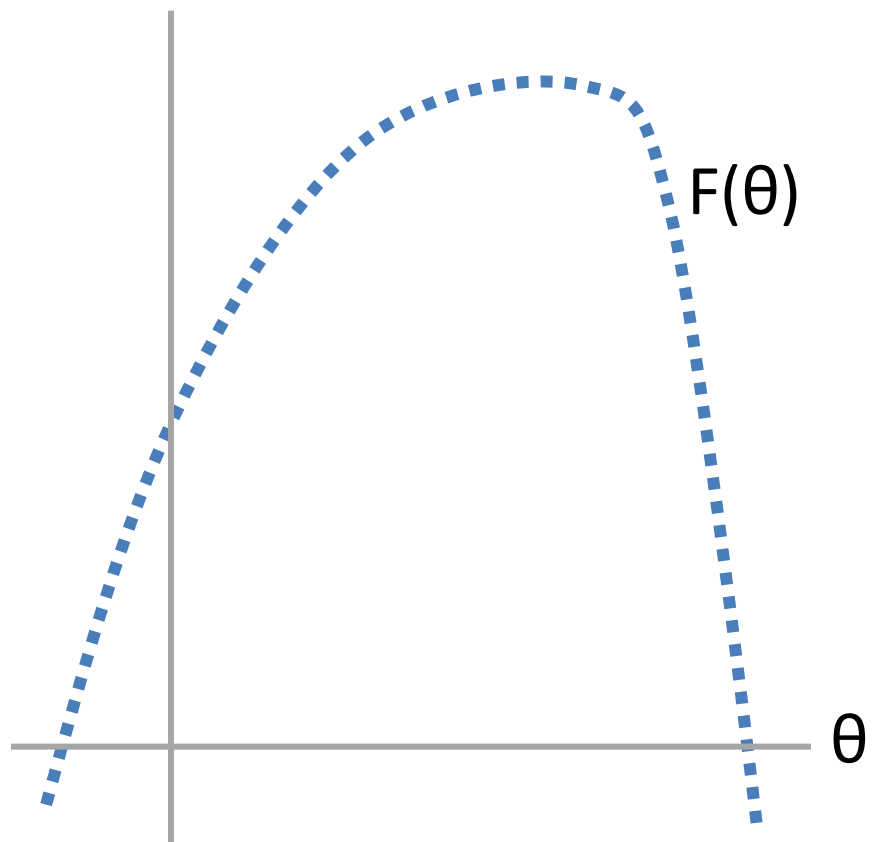
Defining the objective

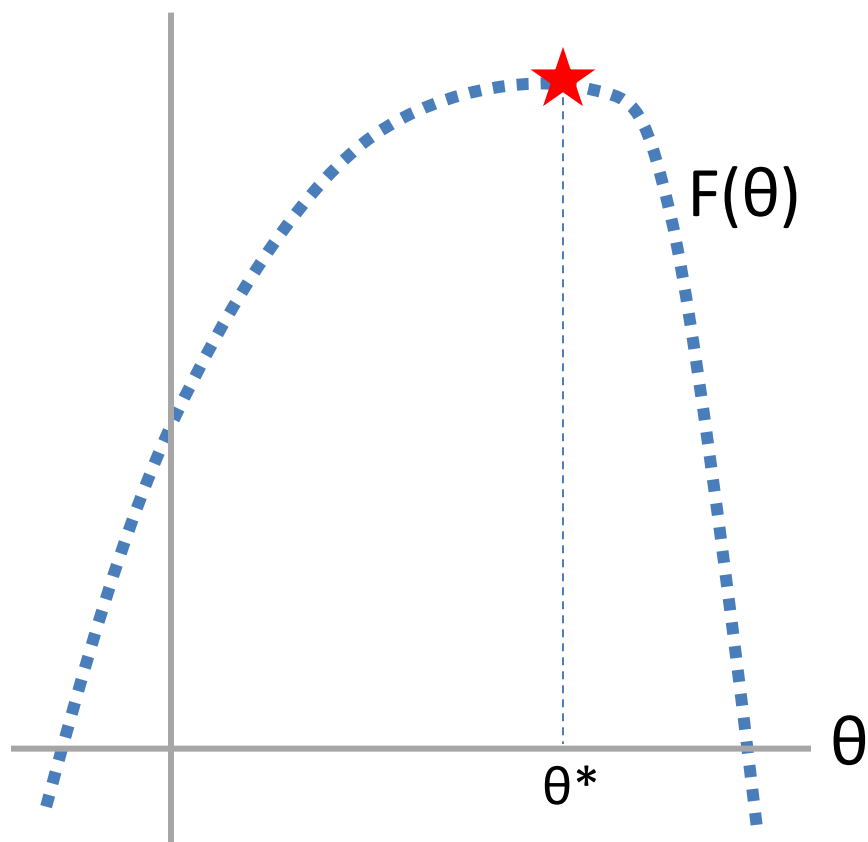
Learning: Optimizing the objective

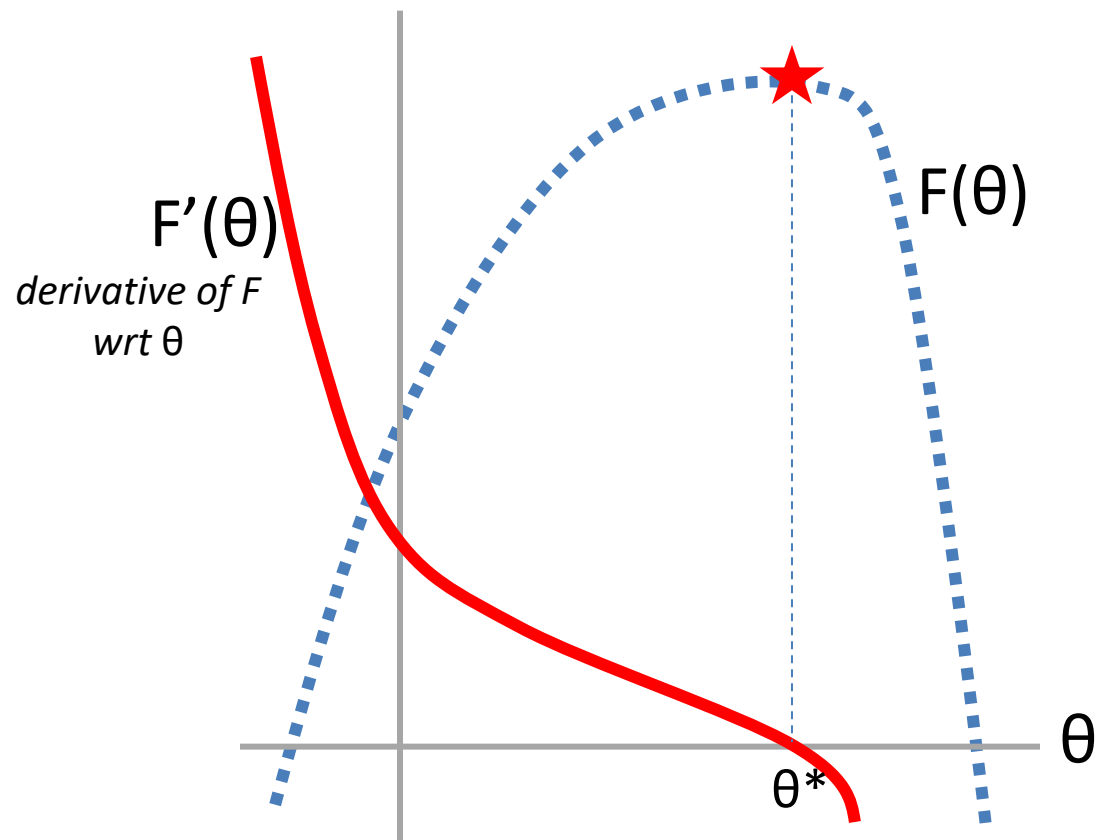
Math: gradient derivation (advanced)

How will we optimize $F(\theta)$?

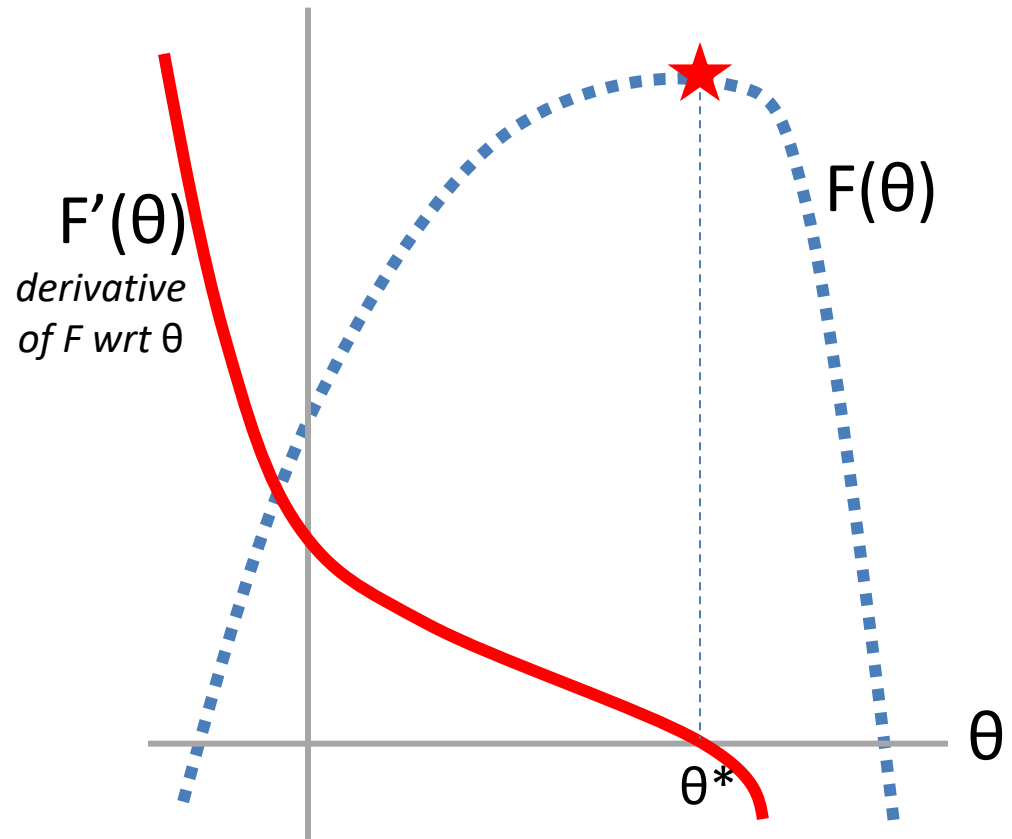
Calculus







What if you can't find the roots?
Follow the derivative



What if you can't find the roots?

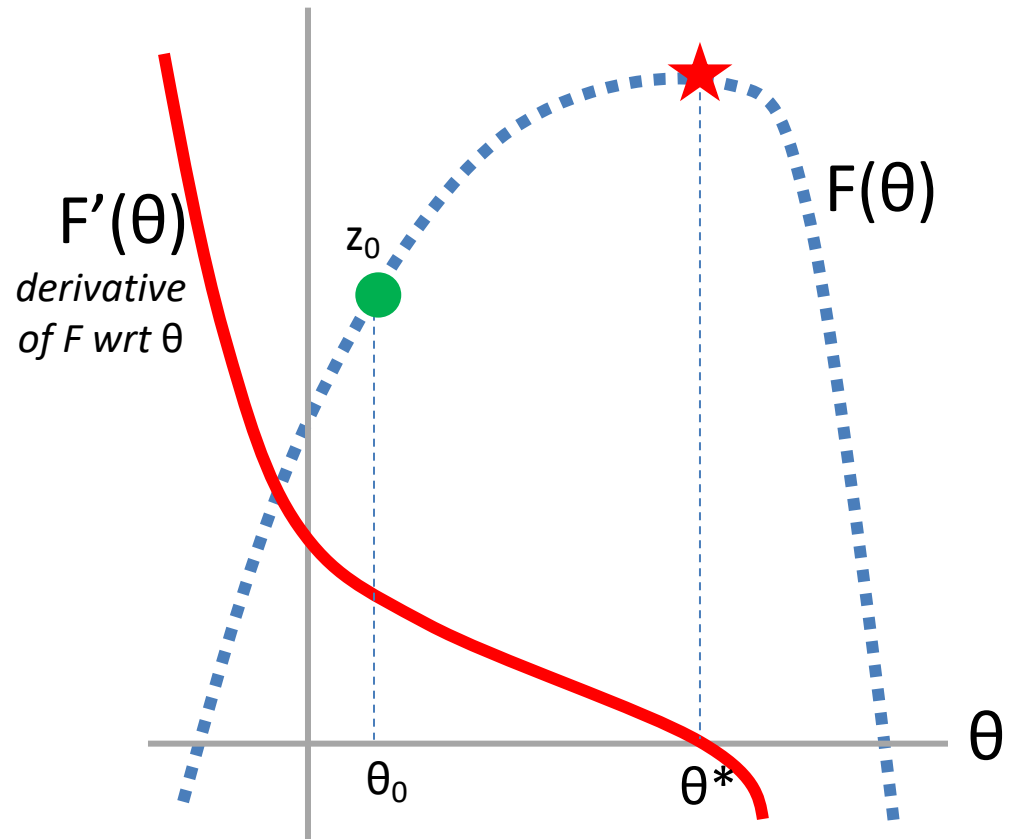
Follow the derivative

Set $t = 0$

Pick a starting value θ_t

Until converged:

1. Get value $z_t = F(\theta_t)$



What if you can't find the roots?

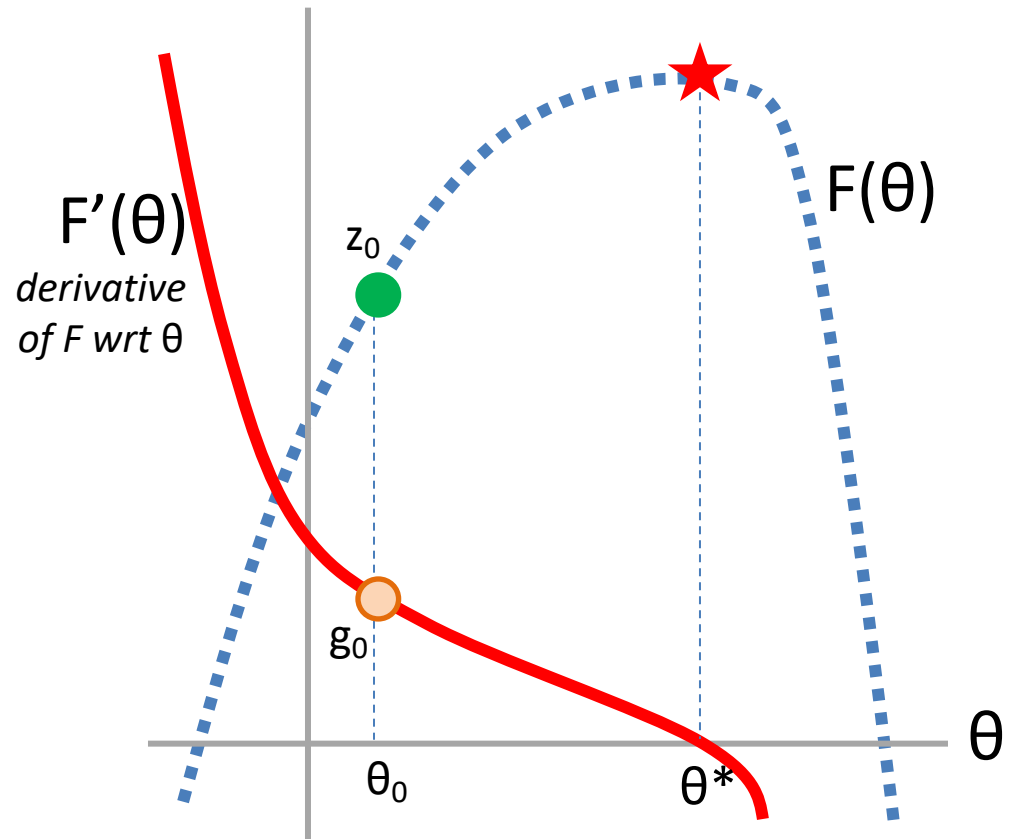
Follow the derivative

Set $t = 0$

Pick a starting value θ_t

Until converged:

1. Get value $z_t = F(\theta_t)$
2. Get derivative $g_t = F'(\theta_t)$



What if you can't find the roots?

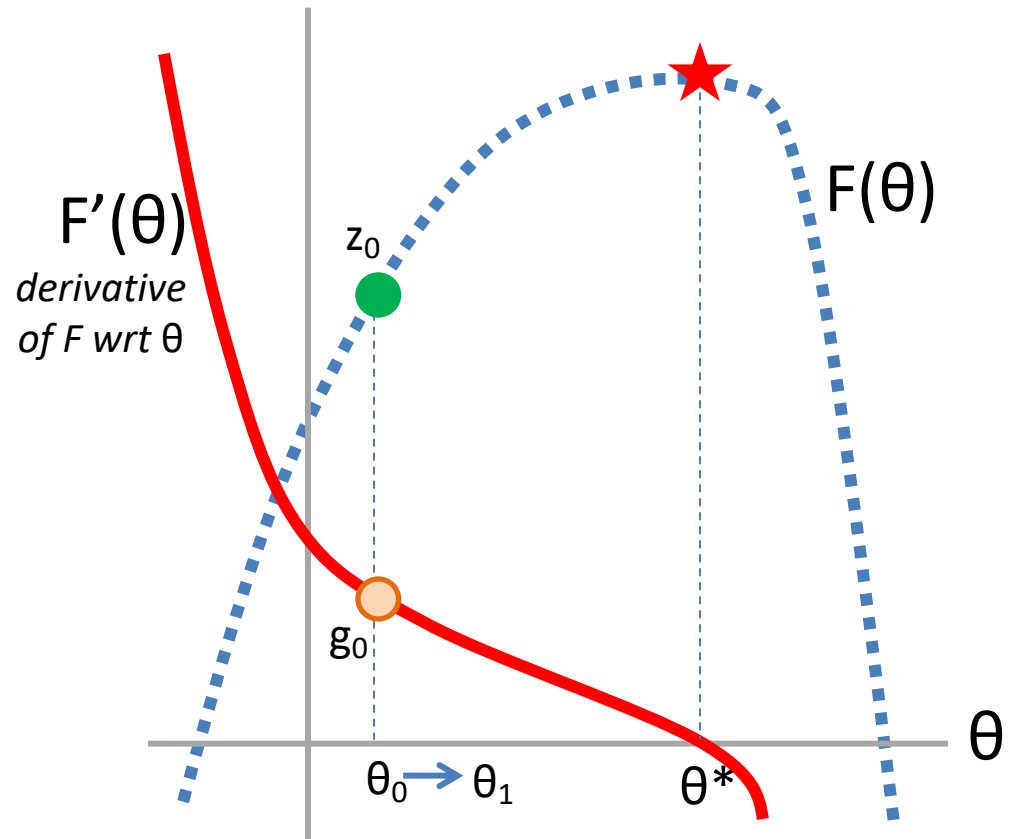
Follow the derivative

Set $t = 0$

Pick a starting value θ_t

Until converged:

1. Get value $z_t = F(\theta_t)$
2. Get derivative $g_t = F'(\theta_t)$
3. Get scaling factor ρ_t
4. Set $\theta_{t+1} = \theta_t + \rho_t * g_t$
5. Set $t += 1$



What if you can't find the roots?

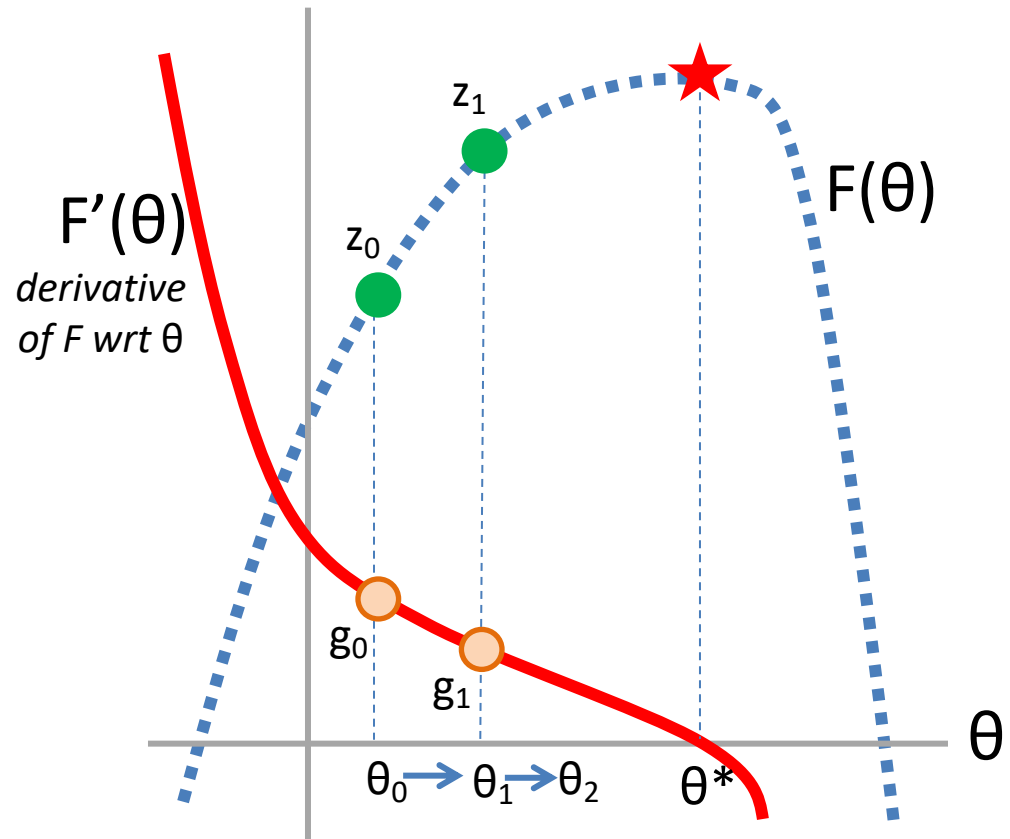
Follow the derivative

Set $t = 0$

Pick a starting value θ_t

Until converged:

1. Get value $z_t = F(\theta_t)$
2. Get derivative $g_t = F'(\theta_t)$
3. Get scaling factor ρ_t
4. Set $\theta_{t+1} = \theta_t + \rho_t * g_t$
5. Set $t += 1$



What if you can't find the roots?

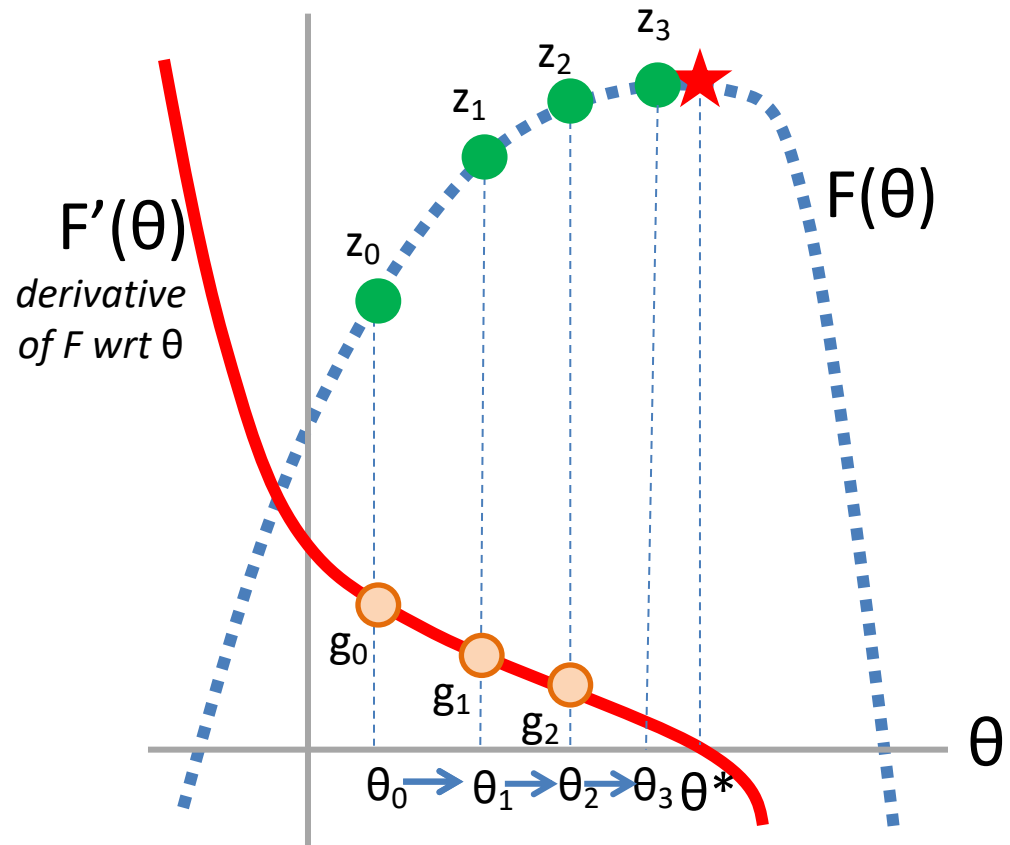
Follow the derivative

Set $t = 0$

Pick a starting value θ_t

Until converged:

1. Get value $z_t = F(\theta_t)$
2. Get derivative $g_t = F'(\theta_t)$
3. Get scaling factor ρ_t
4. Set $\theta_{t+1} = \theta_t + \rho_t * g_t$
5. Set $t += 1$



What if you can't find the roots?

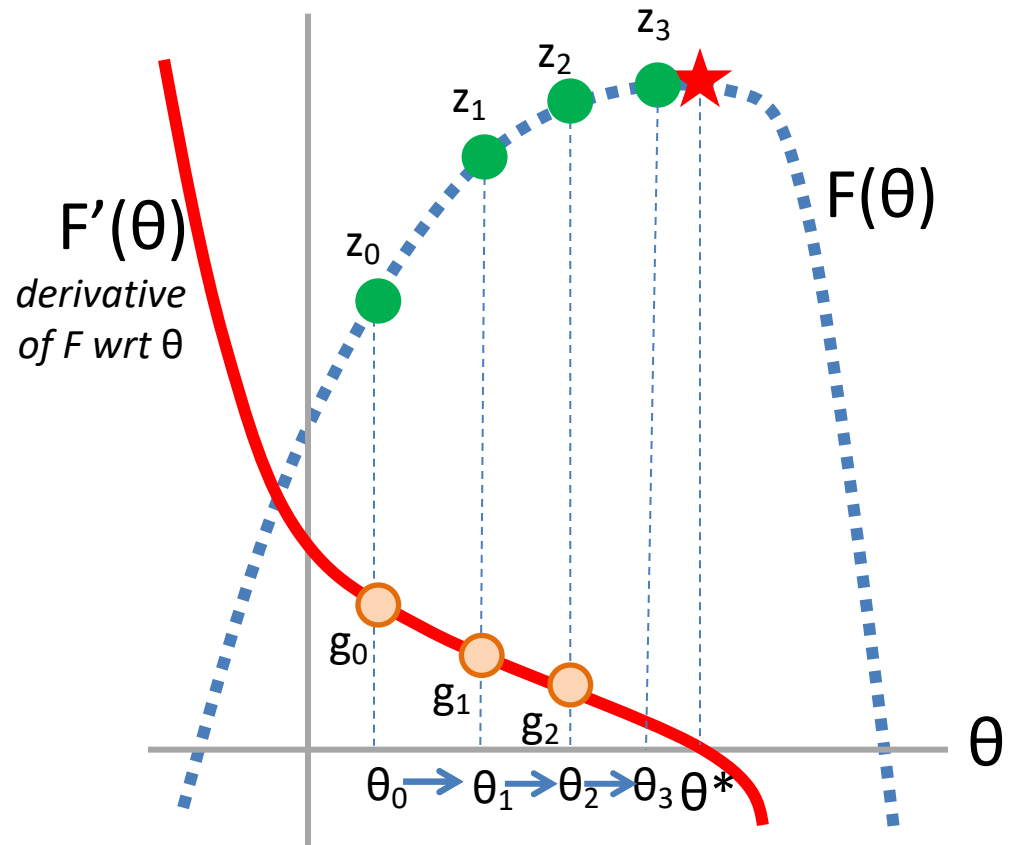
Follow the derivative

Set $t = 0$

Pick a starting value θ_t

Until **converged**:

1. Get value $z_t = F(\theta_t)$
2. Get derivative $g_t = F'(\theta_t)$
3. Get **scaling factor** ρ_t
4. Set $\theta_{t+1} = \theta_t + \rho_t * g_t$
5. Set $t += 1$



Gradient = Multi-variable derivative

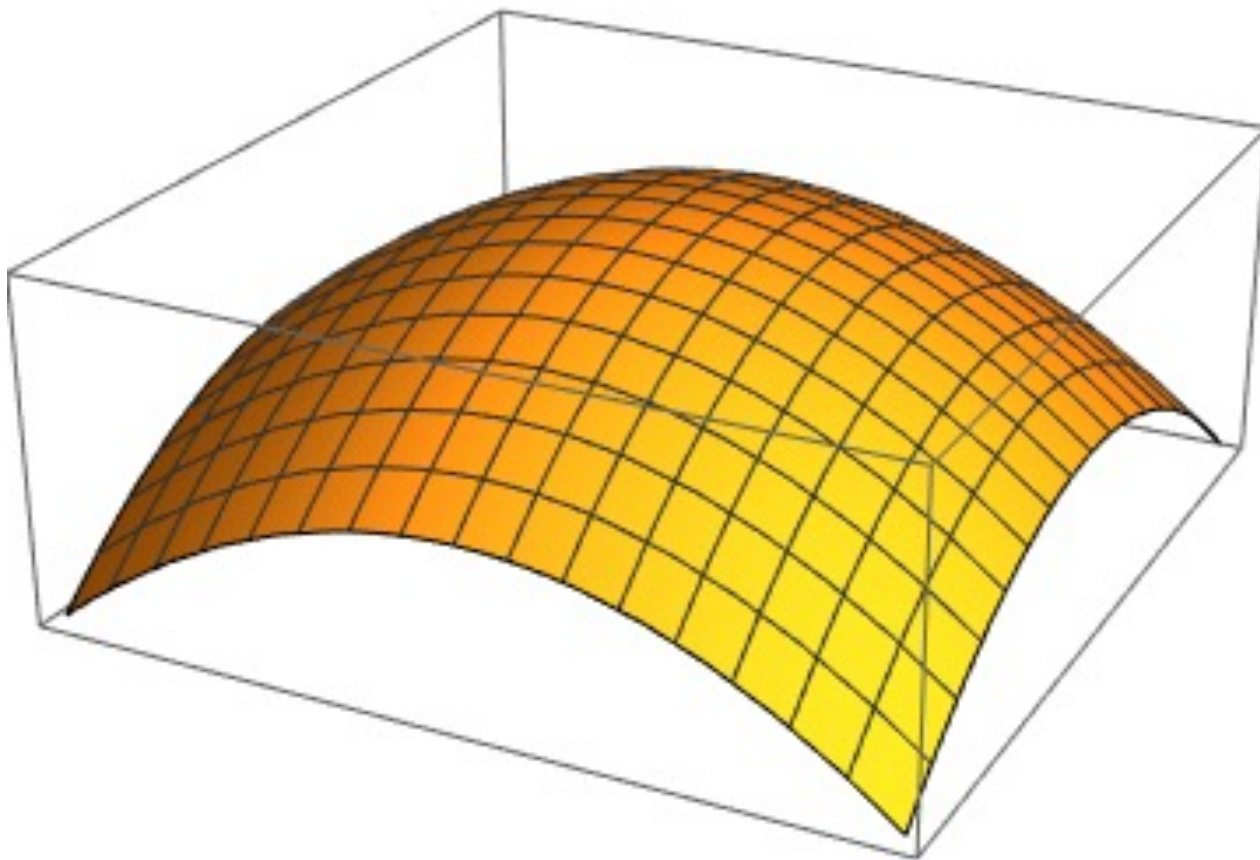
K-dimensional input



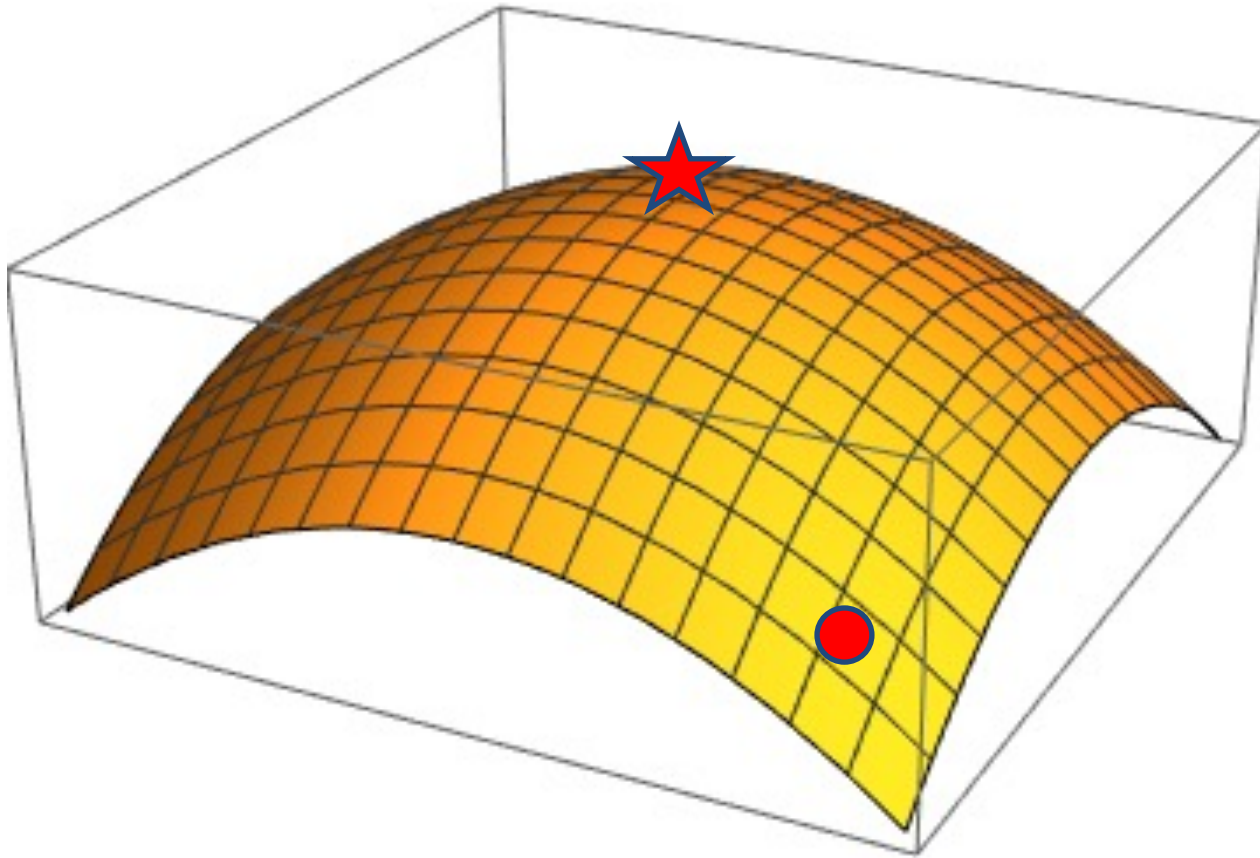
$$\nabla_{\theta} F(\theta) = \left(\frac{\partial F}{\partial \theta_1}, \frac{\partial F}{\partial \theta_2}, \dots, \frac{\partial F}{\partial \theta_K} \right)$$

K-dimensional output

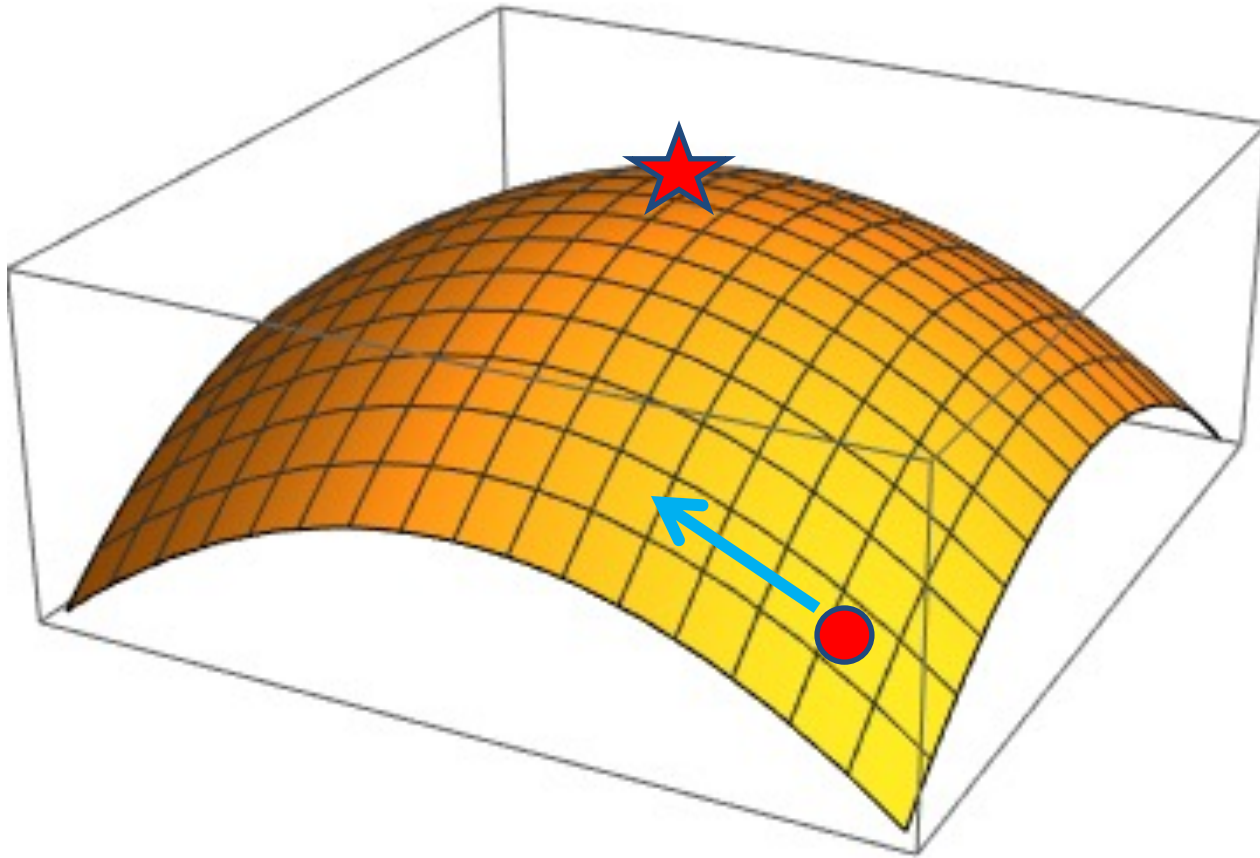
Gradient Ascent



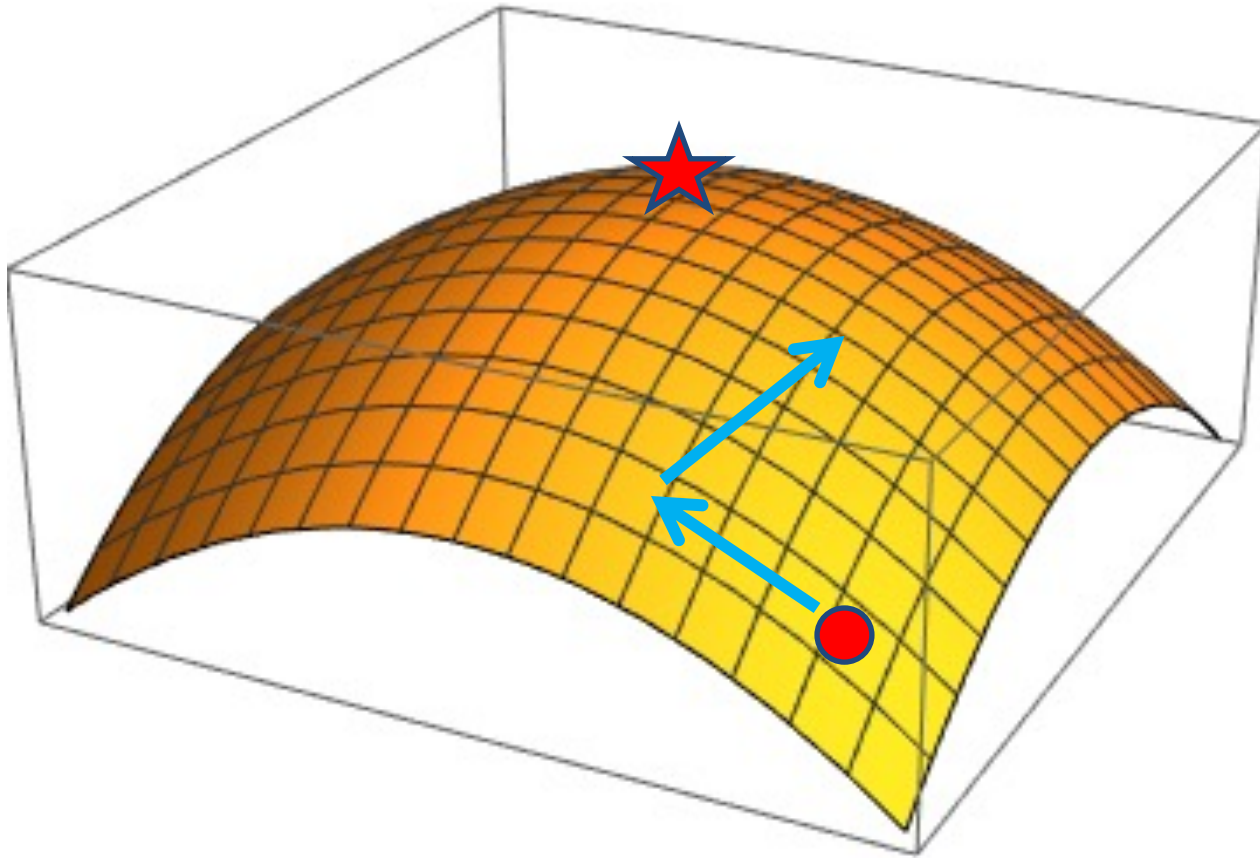
Gradient Ascent



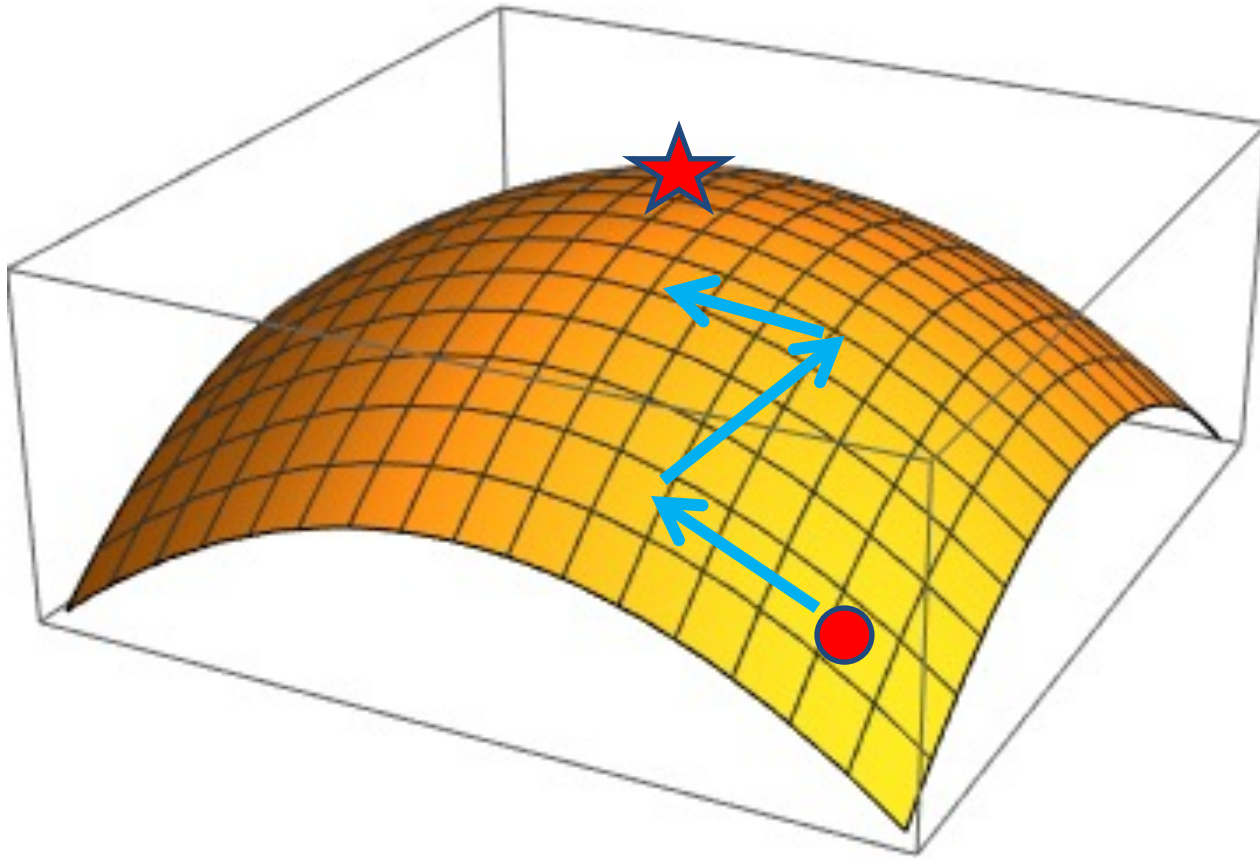
Gradient Ascent



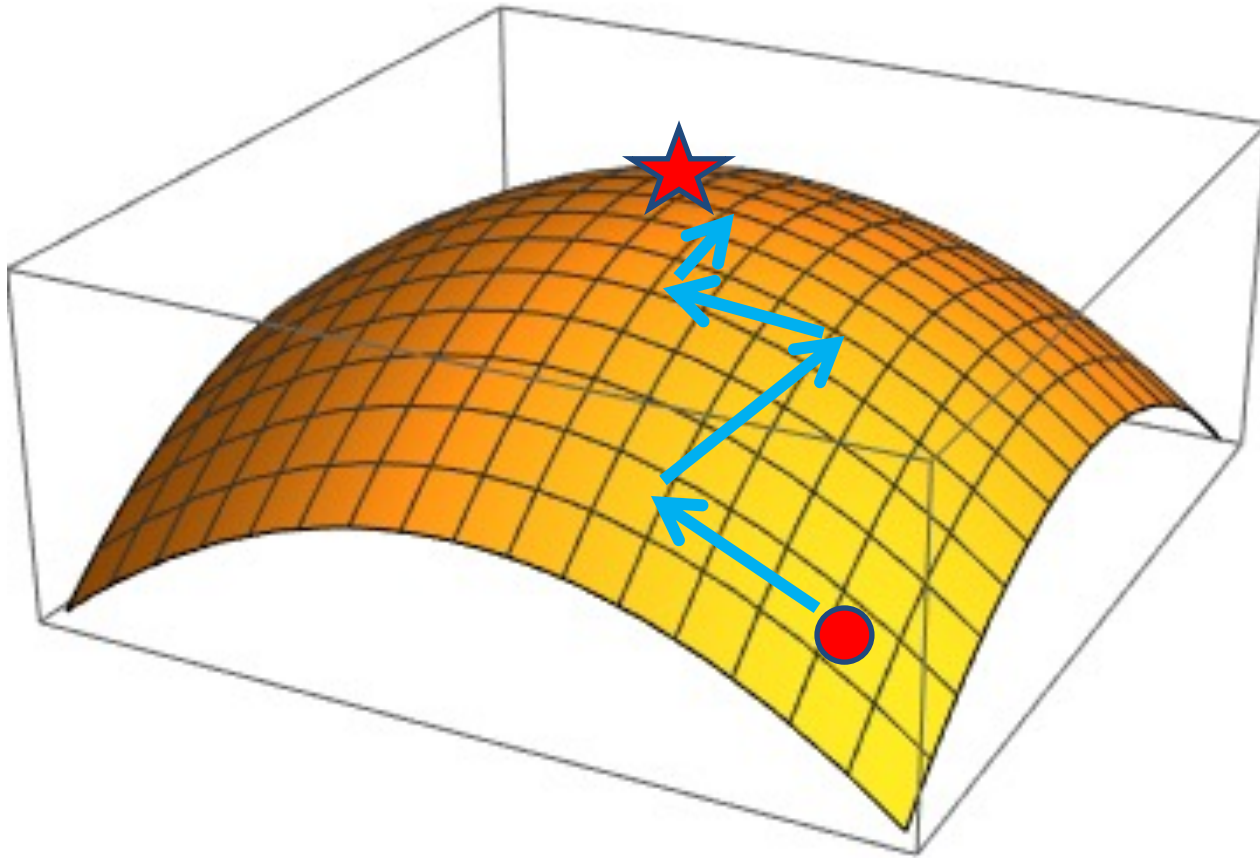
Gradient Ascent



Gradient Ascent



Gradient Ascent



What if you can't find the roots?

Follow the **gradient**

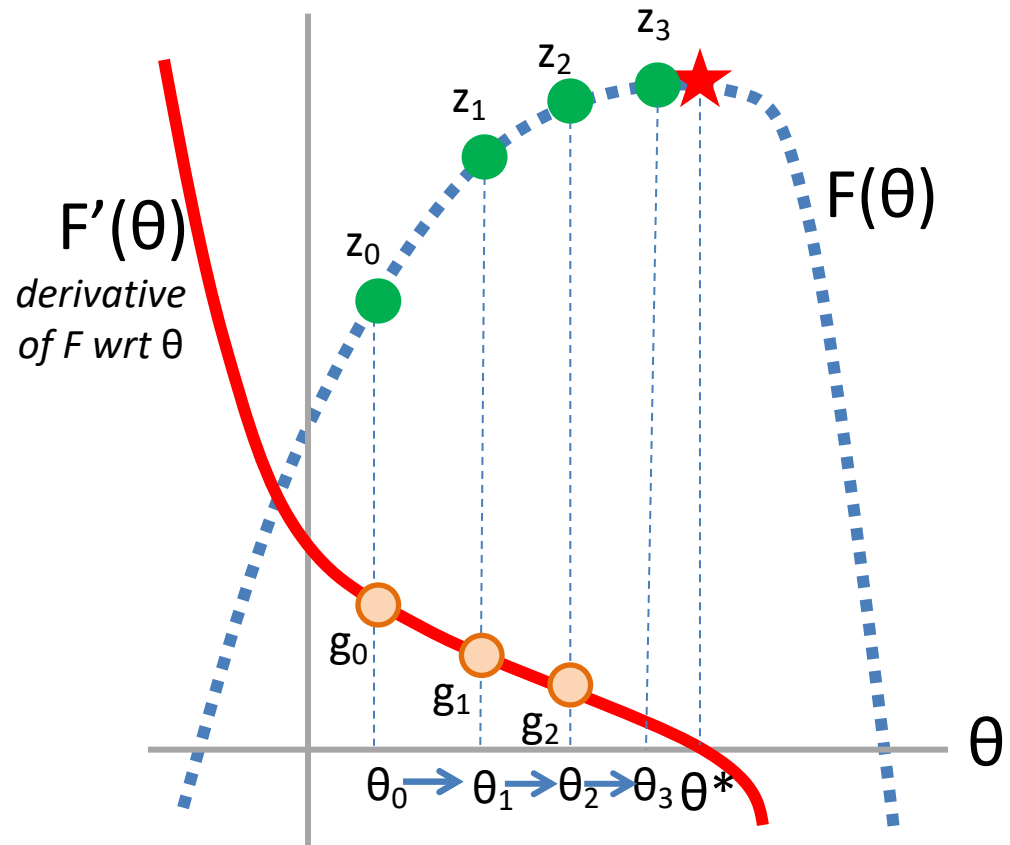
Set $t = 0$

Pick a starting value θ_t

Until converged:

1. Get value $z_t = F(\theta_t)$
2. Get **gradient** $g_t = F'(\theta_t)$
3. Get scaling factor ρ_t
4. Set $\theta_{t+1} = \theta_t + \rho_t * g_t$
5. Set $t += 1$

*K-dimensional
vectors*



Outline

Maximum Entropy classifiers

Defining the model

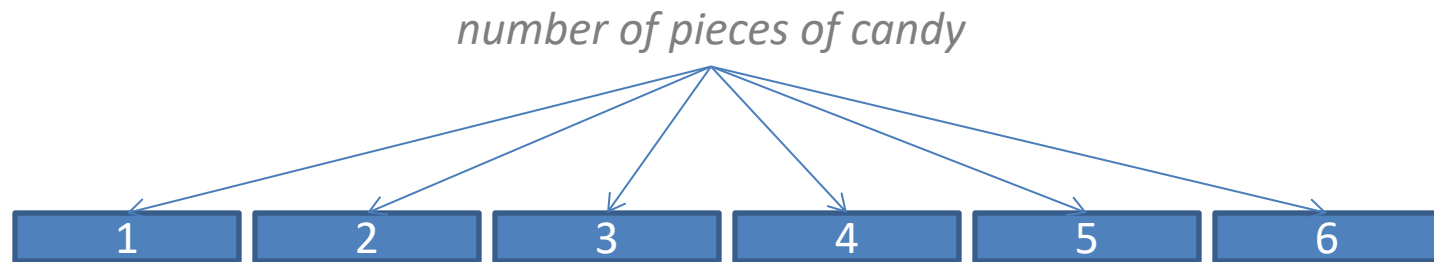
Defining the objective

Learning: Optimizing the objective

Math: gradient derivation (advanced)

Everything after this in this slide deck is “advanced”
(not required, but *highly* recommended for any PhD
or MS thesis student)

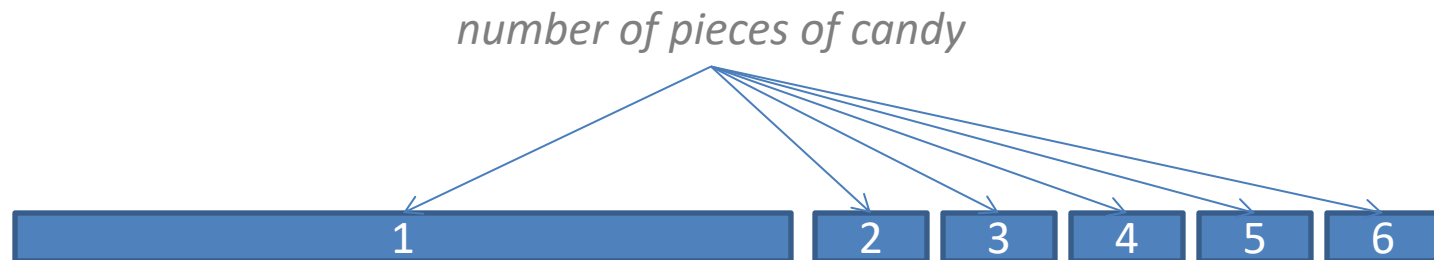
Reminder: Expectation of a Random Variable



$$\begin{aligned} &1/6 * 1 + \\ &1/6 * 2 + \\ &1/6 * 3 + \\ &1/6 * 4 + \\ &1/6 * 5 + \\ &1/6 * 6 \end{aligned} = 3.5$$

$$\mathbb{E}[X] = \sum_x x p(x)$$

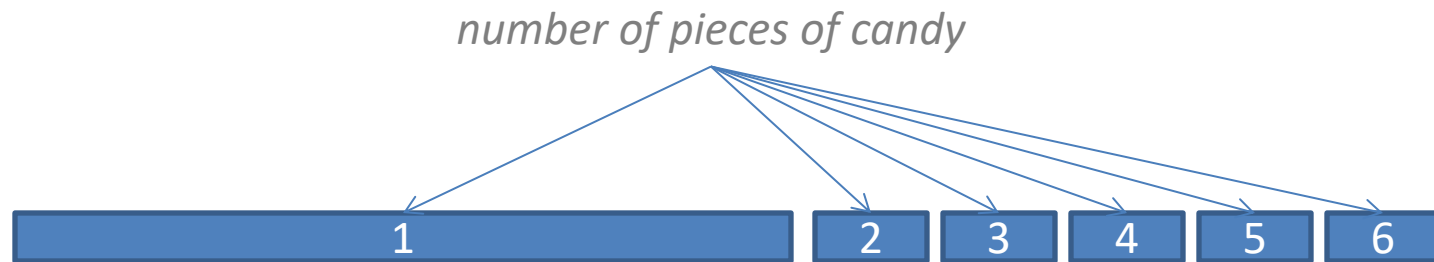
Reminder: Expectation of a Random Variable



$$\begin{aligned} & 1/2 * 1 + \\ & 1/10 * 2 + \\ & 1/10 * 3 + \\ & 1/10 * 4 + \\ & 1/10 * 5 + \\ & 1/10 * 6 \end{aligned} = 2.5$$

$$\mathbb{E}[X] = \sum_x x p(x)$$

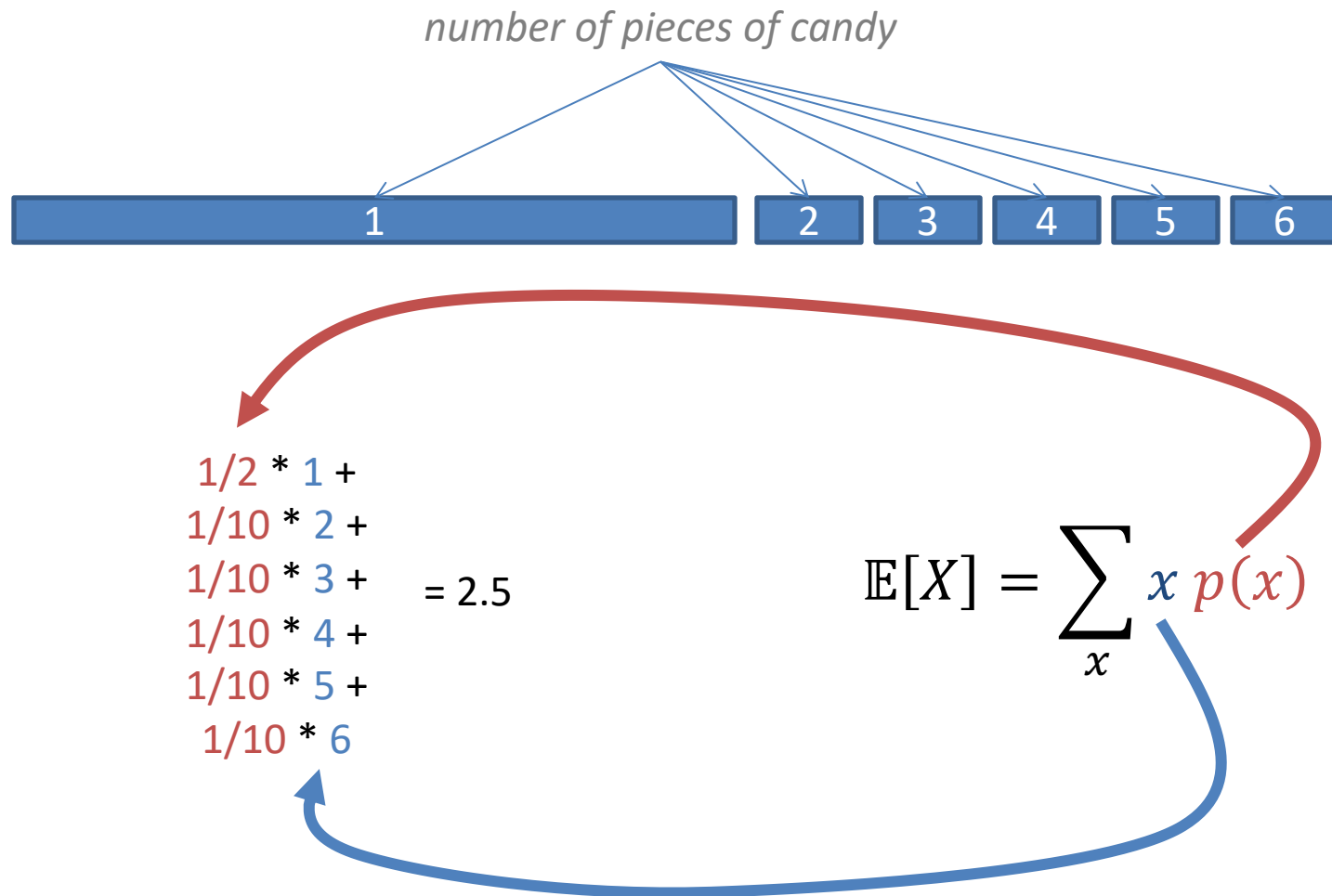
Reminder: Expectation of a Random Variable



$$\begin{aligned} & 1/2 * 1 + \\ & 1/10 * 2 + \\ & 1/10 * 3 + \\ & 1/10 * 4 + \\ & 1/10 * 5 + \\ & 1/10 * 6 \end{aligned} = 2.5$$

$$\mathbb{E}[X] = \sum_x x p(x)$$

Expectations Depend on a Probability Distribution



Log-Likelihood (n-gram LM)

Wide range of (negative) numbers

Sums are more stable

$$\begin{aligned}\log \prod_i p_{\theta}(x_i | h_i) &= \sum_i \log p_{\theta}(x_i | h_i) \\ &= \sum_i \underbrace{\theta^T f(x_i, h_i) - \log Z(h_i)} \\ &= F(\theta)\end{aligned}$$

Differentiating this becomes nicer (even though Z depends on θ)

Log-Likelihood Gradient

Each component k is the difference
between:

Log-Likelihood Gradient

Each component k is the difference
between:

the total value of feature f_k in the
training data

$$\sum_i f_k(x_i, h_i)$$

Log-Likelihood Gradient

Each component k is the difference
between:

the total value of feature f_k in the
training data

$$\sum_i f_k(x_i, h_i)$$

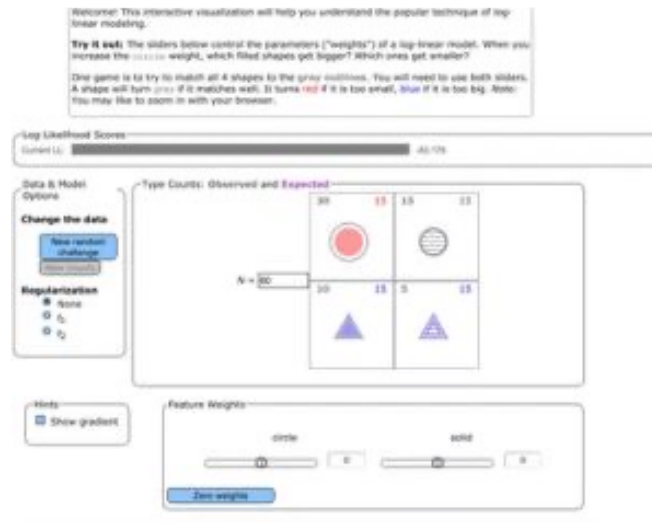
and

the total value the current model p_θ
thinks it computes for feature f_k

$$\sum_i \mathbb{E}_{x' \sim p} [f_k(x', h_i)]$$

<https://www.csee.umbc.edu/courses/undergraduate/473/f21/loglin-tutorial/>

Lesson 6



Authors: Frank Fariello and Jason Eisner.
Find a bug? Want a feature? Think something can be improved? Please file out an [issue report](#).
Last change:

The source code is released [here](#) under the [GNU Affero 3.0 license](#).
The [visual tool](#) and materials are licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](#).
[CC BY-SA](#)
To refer to this work, cite the paper that describes it [\(Fariello & Eisner, 2013\)](#).

Remember: Common Derivative Rules

$$\frac{d \exp x}{dx} = \exp x$$

$$\frac{df(x)g(x)}{dx} = \frac{df(x)}{dx}g(x) + \frac{dg(x)}{dx}f(x)$$

$$\frac{d \log x}{dx} = \frac{1}{x}$$

$$\frac{df(g(x))}{dx} = \frac{df(g(x))}{dg(x)} \frac{dg(x)}{dx}$$

Log-Likelihood Gradient Derivation for LM $p(x_i | h_i)$

$$\nabla_{\theta} F(\theta) = \nabla_{\theta} \sum_i [\theta^T f(x_i, h_i) - \log Z(h_i)]$$

Log-Likelihood Gradient Derivation for LM $p(x_i | h_i)$

$$\nabla_{\theta} F(\theta) = \nabla_{\theta} \sum_i [\theta^T f(x_i, h_i) - \log Z(h_i)]$$

$$= \sum_i f(x_i, h_i) -$$

$$Z(h_i) = \sum_{x'} \exp(\theta \cdot f(x', h_i))$$


Log-Likelihood Gradient Derivation for LM $p(x_i | h_i)$

$$\begin{aligned}\nabla_{\theta} F(\theta) &= \nabla_{\theta} \sum_i [\theta^T f(x_i, h_i) - \log Z(h_i)] \\ &= \sum_i f(x_i, h_i) - \sum_i \sum_{x'} \underbrace{\frac{\exp(\theta^T f(x', h_i))}{Z(h_i)}}_{\text{scalar } p(x' | h_i)} \underbrace{f(x', h_i)}_{\text{vector of functions}}\end{aligned}$$

use the (calculus) chain rule

$$\frac{\partial}{\partial \theta} \log g(h(\theta)) = \left(\frac{\partial g}{\partial h(\theta)} \right) \left(\frac{\partial h}{\partial \theta} \right)$$

scalar $p(x' | h_i)$

vector of functions

Log-Likelihood Gradient Derivation for LM $p(x_i | h_i)$

$$\begin{aligned}\nabla_{\theta} F(\theta) &= \nabla_{\theta} \sum_i [\theta^T f(x_i, h_i) - \log Z(h_i)] \\ &= \sum_i f(x_i, h_i) - \sum_i \sum_{x'} \frac{\exp(\theta^T f(x', h_i))}{Z(h_i)} f(x', h_i)\end{aligned}$$

Do we want these to *fully* match?

What does it mean if they do?

What if we have missing values in our data?

Gradient Optimization for LM

$$p(x_i | h_i)$$

Set $t = 0$

Pick a starting value θ_t

Until converged:

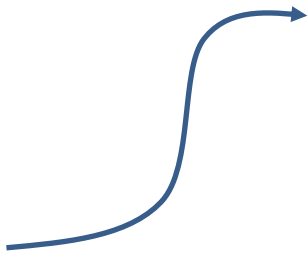
1. Get value $z_t = F(\theta_t)$

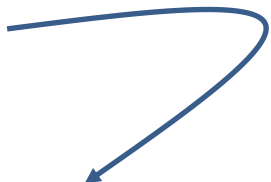
2. Get derivative $g_t = F'(\theta_t)$

3. Get scaling factor ρ_t

4. Set $\theta_{t+1} = \theta_t + \rho_t * g_t$

5. Set $t += 1$


$$\sum_i \theta^T f(x_i, h_i) - \log Z(h_i)$$


$$\frac{\partial F}{\partial \theta_k} = \sum_i f_k(x_i, h_i) - \sum_i \sum_{x'} f_k(x', h_i) p(x' | h_i)$$

Gradient Optimization for Classifier

$p(y | \mathbb{X})$

Set $t = 0$

Pick a starting value θ_t

Until converged:

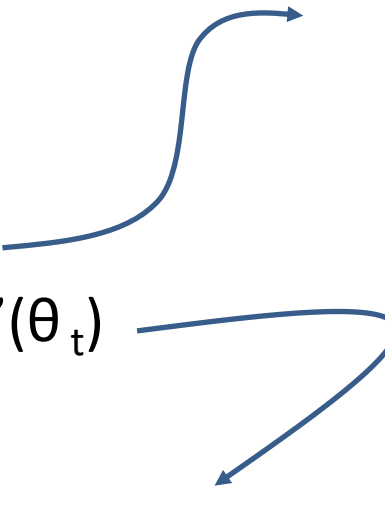
1. Get func. value $F(\theta_t)$

2. Get derivative $g_t = F'(\theta_t)$

3. Get scaling factor ρ_t

4. Set $\theta_{t+1} = \theta_t + \rho_t * g_t$

5. Set $t += 1$

$$\theta^T f(\mathbb{X}, y) - \log Z(\mathbb{X})$$


$$\frac{\partial F}{\partial \theta_k} = f_k(\mathbb{X}, y) - \sum_{y'} f_k(\mathbb{X}, y') p(y' | \mathbb{X})$$

Outline

Maximum Entropy classifiers

Defining the model

Defining the objective

Learning: Optimizing the objective

Math: gradient derivation (advanced)