EXHIBIT 1

The Official Google Blog
Insights from Googlers into our products, technology, and the Google culture.

Personalized Search for everyone

12/04/2009 03:01:00 PM

Today we're helping people get better search results by extending Personalized Search to signed-out users worldwide, and in more than forty languages. Now when you search using Google, we will be able to better provide you with the most relevant results possible. For example, since I always search for [recipes] and often click on results from epicurious.com, Google might rank epicurious.com higher on the results page the next time I look for recipes. Other times, when I'm looking for news about Cornell University's sports teams, I search for [big red]. Because I frequently click on www.cornellbigred.com, Google might show me this result first, instead of the Big Red soda company or others.

Previously, we only offered Personalized Search for signed-in users, and only when they had Web History enabled on their Google Accounts. What we're doing today is expanding Personalized Search so that we can provide it to signed-out users as well. This addition enables us to customize search results for you based upon 180 days of search activity linked to an anonymous cookie in your browser. It's completely separate from your Google Account and Web History (which are only available to signed-in users). You'll know when we customize results because a "View customizations" link will appear on the top right of the search results page. Clicking the link will let you see how we've customized your results and also let you turn off this type of customization.

Check out our help center for more details on personalized search, how we customize results and how you can turn off personalization. Learn more by watching our video:

Posted by Bryan Horling, Software Engineer and Matthew Kulick, Product Manager

Permalink

Share:

Labels: search

Links to this post

| When Google Narrows Your Search

| What is Google Personalized Search

| Keres•marketing helyzet 2010 – SEMPO adatok

| The Rise of Universal Paid Search

| Google Uses Personal Data to Tailor 20% of Searches

| Thank you Google for this precious lesson about privacy

| Dumb SEO is DEAD

| g**gl*s personalisierte Suche nun auch für ausgeloggte Benutzer

| Google Buzz: Convenience or the ultimate data tracking tool?

| Google Search Gets Personal

| Jag är tillbaka i SEO-världen

| Is your SEO Really Optimized?

| Why did Google lumber us with Personalised Search?

| Is Google making us less rational?

| How To Fight Personalization — Is It Possible?

| Historique et personnalisation des résultats sur Google

| Personalized Results and Paid Search Are Not A Match

| Keeping Score: 10 Predictions for 2009 — How'd Bruce Do?

| Google-Blues: Wo bleibt der Long-Tail-Effekt?

| Google in Review: The Search Giant in 2009

| Personalized Search And Your SEO Efforts

| Personalized Search And Your SEO Efforts

| Reader Rescue: Do I have to disable Google personalization ...

Google Public Policy Blog

Google's Innovation Factory (and how testing adapts)

Google Testing Blog

About Google Code Jam 2010

Google Student Blog

Blurring the Line: Non-Line Driven Analytics

Google Retail Blog

Newest Google blogs

- DoubleClick for Publishers API Blog
- Google Translate Blog
- Google Wave Blog
- Google New Zealand Blog
- Data Liberation Blog

Labels

- accessibility (27)
- acquisition (11)
- ads (69)
- Africa (2)
- apps (298)

| Q and A: Do I have to disable Google personalization settings to ...

| Q and A: Do I have to disable Google personalization settings to ...

| Is Google Watching You?

| Personnalisation des résultats Google, pensez à désactiver l ...

| Don't Trust that Google Ranking: The New Personalized Search Blindfold

| Google Personalizes Search Results for Everyone

| Google's Changes and What You Need To Do On Today on Ecom Experts

| Personalized search by default

| 10 News Media Content Trends to Watch in 2010

| 10 News Media Content Trends to Watch in 2010 | Vadim Lavrusik ...

| 10 News Media Content Trends to Watch in 2010

| Is the only protectionist ORM strategy one that embraces optimised ...

| Advent Calendar – Time to reflect: social media and world events

| Google stories

| How Personalized Search Changes SEO (and Doesn't)

| How Personalized Search Changes SEO (and Doesn't)

| El apabullante -y escalofriante- ritmo de Google

| Google Takes it Personal

| AdWords-Shortcuts: News im Überblick KW 51

| “SEOは死んだ”……一部のSEO業者にとっては など10記事（海外&国内SEO ...

| Your Google results are about to get weirder

| _____

|

|

|

|

|

|

|

|

| Big Changes at Google

| Can SEO Exist Beyond Google Personalization?

| Can SEO Exist Beyond Google Personalization?

| Google now defaults to Personalized Search for everyone

| Will the middle be harder to find?... As tenor and temperature of ...

| Google Is “Personalizing” Searches Here's How To Get Real Ranking

| The Weekly Insider 12-7-09 to 12-11-09

| Google vs SEO

| Персонализирано търсене – няма “не искам”

| Personalized search is now Google's default

| Персонализирано търсене за всички

| Top 5 Reasons for Ongoing SEO Services

| Google : La recherche personnalisée devient universelle

| Real Time Search Has Arrived!

| Real Time Search Has Arrived!

| Google blir personligt som standard

| Google's SEO Bombshell - Personalized Search is Here

| All You Need to Know About Google's New Feature Updates

| ヤフーのアルゴリズム更新でまた被害者続出! など10記事（海外&国内SEO ...

| Golpe al SEO: búsqueda personalizada por defecto para todos

| Personalisierung von Suchergebnissen - fünf Probleme!

| Google Personalised Search

| Google : résultats personnalisés pour tout le monde… So what ?

| Contextualised Relevance

| New Google Tools Promising for Affiliates

| Google Personalizes Search Results for Everyone

| Google: Chrome un paplašinājumi, kā arī daudz kas cits

| Google blir personlig - information och instruktioner

| Google utökar personaliserat sök till alla

| Wijziging Google Personalized Search – de gevolgen

| •••• ••••• •••• "••••• ••••••" •••••• ••••• ••••••

| Залогинен-разлогинен, получаю свою порцию персональных результатов

| Google's Latest Releases Part 1: Real-time Search

| Google Extends Personalized Search

| Personalized Search: The End for Some, but a Breakthrough for Most ...

| Google's Universal Customization Has SEO Implications

| Google Personalised Search

| Personalized search for all at Google

| Personalizará Google los Resultados de los Dispositivos?

| Más personalización, nuevos desafíos

| Endgame/New Game: Google Search Moves Focus on The Moment of ...

| How will Google Personalized Search affect affiliate marketing?

| Don't Be Evil

| Google announcements - personal search, visual search, realtime ...

| All Google Searches Are Now Automatically 'Personalized'

| GOOGLE: orice utilizator este urmarit in numele unui serviciu mai bun

| Popeye, Goggles und mehr: Google dreht auf

| Personalized Search Opacity

| La ricerca personalizzata di Google e le possibili ricadute nel ...

| 10 viktiga konsekvenser med personliga sökresultat

| SEOの終焉！？ ―Googleのパーソナライズ検索

| Google is watching you!

| PsychoCoder's Realm - Google has crossed the line

- Mobile
- More...

What We're Reading

- Ars Technica
- Ask.com Blog
- Google Guide
- Google OS
- John Battelle's Searchblog
- Marketing Pilgrim
- MSN Search Weblog
- O'Reilly Radar
- Pandia Search World
- Philipp Lenssen's Google Blogoscoped
- Read/Write Web
- Search Engine Journal
- Search Engine Land
- Search Engine Roundtable
- Search Engine Watch Blog
- Slashdot - Google
- Techdirt
- The Launch Pad - X PRIZE
- Traffick
- WebmasterWorld
- Yahoo Search Blog

| Google Presents Personalized Search for Everyone

| Google and the Ostrich Effect

| Google personalizará las búsquedas aunque el internauta no lo pida

| Google, how far is too far? in Corner Cubicle

| Google har blivit personligt – alldeles otroligt personligt

| The Implication of Google Personalized Results

| Personalizacja wyników wyszukiwania i usage data w SEO

| Google: Recherche en Temps Réel et Résultats Personnalisés

| Code 2D, recherche temps réel, recherche visuelle et personnalisée...

| Google Offers 'Personalized Search' To Everyone

| Google – Inflacija novotarija. Fokus: Social Search

| From "Personalized Search for everyone" to "Personalized ...

| A Final Nail in the Coffin of “Google Ground Truth”?

| Google offers personalised search even when not logged into Google

| The new search engine: the impact of personalized search results

| Eric Schmidt, The Wall Street Journal and Personalized Search

| Expanded Google Personalized Search…What Does It Mean for SEO?

| Google personalizará las búsquedas aunque el internauta no lo pida

| Google Personalized Search

| Personalisierte Suche für jedermann

| Google personalizzato di default

| Google, The Big Brother

| Google, The Big Brother

| Google: I See Results You Don't See...?

| Google généralise la personnalisation des résultats

| Google personalizará las búsquedas aunque el internauta no lo pida

| TechCrunch: Lazyfeed's New Realtime Interface Tips Into ...

| Google Will "Personalize" Your Search When You're Not Logged In

| Google personalises everyone's search results

Blogs By Googlers

- 20-Something Finance
- Abe Tries Again
- Alon Chen's Diary-•••• ••
- Beyond Satire
- Bikin' my Bloggin'
- Bladam 2.0
- Bolinfest Changeblog
- Catspaw's Guide to the Inevitably Insane
- Confessions of a Digital Packrat
- Damon Kohler
- Donal Mountain's notes
- Dr. Razavi's Good-to-Know-Info
- Ego Food
- Erica's Joys
- Germart
- Grokster
- Gyula Simonyi: smart design
- iBanjo
- It Has Come to My Attention
- Jason Morrison.net
- Jens Meiert
- JR Says
- Kraneland
- Lorem Ipsum

'

| Lazyfeed's New Realtime Interface Tips Into Information Overload

| The SEO guide to Google personalized search

| Google extends personalised search to all users

| Can You "Rank" in Google if Everyone Has Different Search Results?

| Google Search Gets Personal With Everyone

| Google / Персонализация для анонимов

| Google extends personalised search to all users

| El problema de la personalización inevitable

| Problems with Google's personalized search

| Personalized search at Google

| Positionnement Google : la personnalisation va t-elle changer la ...

| Google lance la recherche personnalisée pour tous

| Google Extended Personalized Search for Everyone

| Getting To Know You, Getting To Know All About You

| Google Tracks & Personalizes Even When You're Signed-Out

| Google and its new personalized search

| The Importance of Social Networking Just Grew Exponentially Overnight

| Google Altered SERPS For EVERYONE Now

| ••• ••••••• •••• •••••• 12/4/2009

| Google Roundup

| Google Personalised Search

| Google's personalized search now works even when you're not signed in

| Fluch oder Segen? Personalisierte Suchergebnisse bei Google

| Google personalisiert alle Suchanfragen

| Personalised Search

| Google da a cada uno un resultado

| Google Caffeine Update

| Gepersonaliseerde zoekresultaten Google zonder in te loggen

|

.

| Personalized Search for everyone (Bryan Horling/The Official ...

| Personalized Search and SEO

Create a Link

Newer Post | Home | Older Post

EXHIBIT 2

Google Ads Preferences

Make the ads you see on the web more interesting

Many websites, such as news sites and blogs, partner with us to show ads on their sites. To see ads that are more related to your interests, edit the interest categories below, which are based on sites you have recently visited. Learn more

Your interests are associated with an advertising cookie that's stored in your browser. If you don't want us to store your interests, you can opt out below.

Watch our video:
Ads Preferences explained

Ads Preferences affect ads that Google shows on other websites.

Your interests

Below you can edit the interests that Google has associated with your cookie:

Category
Business - Advertising & Marketing
Computers & Electronics - Software - Operating Systems - Mac
Internet - Search Engine Optimization & Marketing
Internet - Web Services - Search Engines
Lifestyles - Activism & Social Issues
Local - Regional Content
News & Current Events - Newspapers

Add interests Google does not associate sensitive interest categories with your ads preferences.

Opt out

Opt out if you prefer ads not to be based on the interest categories above.

Opt out

When you opt out, Google disables this cookie and no longer associates interest categories with your browser.

Your cookie

Google stores the following information in a cookie to associate your ads preferences with the browser you are currently using:

```
id=2207ad96f9000012|2588783/907846/14781,1092688/350444/14781,2
```

Visit the Advertising and Privacy page of our Privacy Center to learn more.

Google is a participating member of the Network Advertising Initiative.
You can opt out of this cookie as well as other network advertising cookies from the Network Advertising Initiative opt-out page.

Your ads preferences only apply in this browser on this computer. They are reset if you delete your browser's cookies.

Exhibit 3

Recommended Stories: How it works

When you sign in to personalized News and keep Web History enabled, you allow Google to track and save your news selections. Then, Google News can automatically recommend relevant stories just for you by using smart algorithms that analyze your selections.

The News algorithms compare your tastes to the aggregate tastes of other groups of Google News users. Simply put, we recommend news stories to you that have been read by many other users who've also read similar stories as you in the past. The more you use Google News while you're signed in to your Google Account, the better your recommendations will become over time.

Learn more about Web History. **Note:** we can't provide recommended news for you if you don't sign in to your Google Account, or if you turn off the Web History component of personalized Google News.

Recommended articles

- Recommended Stories: Basics
- News for Mobile Devices: Supported Phones and Devices
- How it Works: Languages and regions
- Content in Google News: Feedback on articles
- Content in Google News: Weather information

Learn from other Google users

Find answers, ask questions, and share your expertise with others in the Google News Help Forum.

Help for Publishers

Are you a news publisher? We encourage you to visit our Publisher Help Center for help with your site. Here, you'll find our most comprehensive, up-to-date information for publishers.

Recently, on the official News Blog

7/15/2010
Posted by Chris Beckmann, Product Manager

Two weeks ago we gave the Google News homepage a new look and feel with enhanced customization, discovery and sharing. This
Read more

Suggest A Feature

Got a feature suggestion for Google News? Let us know

EXHIBIT 4

THE AMERICAN HERITAGE DICTIONARY OF THE ENGLISH LANGUAGE

All correspondence and inquiries should be directed to
Dictionary Division, Houghton Mifflin Company
One Beacon Street, Boston, Massachusetts 02107

ISBN: 0-395-20360-0 (new college edition; thumb-indexed)
0-395-20359-7 (new college edition; plain edges)
0-395-24575-3 (high-school edition)

Library of Congress Catalog Card Number 76–86995

Manufactured in the United States of America

Computer-composed by Inforonics, Inc.
in Maynard, Massachusetts

es·ter·i·fi·ca·tion (ĕ-stĕr′ə-fə-kā′shən, ĭ-stĕr′-) *n.* Any reaction resulting in the formation of at least one ester product.

es·ter·i·fy (ĕ-stĕr′ə-fī′, ĭ-stĕr′-) *v.* **-fied, -fying, -fies.** —*intr.* To change to an ester. —*tr.* To change (a compound) into an ester. [ESTER + -FY.]

Es·ther[1] (ĕs′tər). A feminine given name. [Greek *Esthēr*, from Hebrew *'Estēr*, from Persian *sitareh*, star. See **ster-**[3] in Appendix.*]

Es·ther[2] (ĕs′tər) *n.* A book of the Old Testament recounting the story of Esther, the Jewish queen of Persia who saved her people from massacre.

es·the·sia (ĕs-thē′zhə, -zhē-ə) *n.* The ability to receive sense impressions. [New Latin, back-formation from ANESTHESIA.]

es·thet·ics (ĕs-thĕt′ĭks) *n.* **Aesthetics** (*see*). —**es′thete′** (-thēt′) *n.* —**es·thet′ic** *adj.* —**es′the·ti′cian** *n.* —**es·thet′i·cism′** *n.*

Es·tho·ni·a. See **Estonia.**

Es·tho·ni·an. Variant of **Estonian.**

Es·tienne (ĕs-tyĕn′). French family of printers and publishers, including **Henri** (1460?–1520) and his sons, **François** (1502–1550), **Robert** (1503–1559), and **Charles** (1504–1564).

es·ti·ma·ble (ĕs′tə-mə-bəl) *adj.* **1.** Capable of being estimated or evaluated; calculable. **2.** Deserving of esteem; admirable. —**es′ti·ma·ble·ness** *n.* —**es′ti·ma·bly** *adv.*

es·ti·mate (ĕs′tə-māt′) *tr.v.* **-mated, -mating, -mates.** **1.** To make a judgment as to the likely or approximate cost, quantity, or extent of; calculate approximately. **2.** To form a tentative opinion about; evaluate: *"While an author is yet living we estimate his powers by his worst performance"* (Samuel Johnson). —See Synonyms at **calculate.** —*n.* (ĕs′tə-mĭt). *Abbr.* **est.** **1.** A tentative evaluation or rough calculation. **2. a.** A preliminary calculation submitted by a contractor or workman of the cost of work to be undertaken. **b.** The written statement of such a calculation. **3.** A judgment based upon one's impressions; an opinion. [Latin *aestimāre*.] —**es′ti·ma′tive** *adj.* —**es′ti·ma′tor** (-mā′tər) *n.*

Synonyms: *estimate, appraise, assess, assay, evaluate, rate.* These verbs mean to form a judgment of worth or significance. *Estimate* may imply judgment based on rather rough calculation. In general it lacks the definitiveness of the other terms, especially *appraise*, which stresses expert judgment. *Assess* implies authoritative judgment; it involves setting a monetary value on something as a basis for taxation. *Assay* likewise refers to careful examination, such as chemical analysis of ore to determine its content. In extended senses, *appraise, assess,* and *assay* can refer to any critical analysis or appraisal. *Evaluate* implies considered judgment in setting a value on a person or thing. *Rate* involves determining the rank of a person or thing when he or it is judged in relation to others of the same kind.

es·ti·ma·tion (ĕs′tə-mā′shən) *n.* **1.** The act or an instance of estimating. **2.** An opinion reached by estimating; judgment: *"No man ever stood the lower in my estimation for having a patch in his clothes"* (Thoreau). **3.** Favorable regard; esteem.

es·ti·val. Variant of **aestival.**

es·ti·vate. Variant of **aestivate.**

es·ti·va·tion. Variant of **aestivation.**

Es·to·ni·a (ĕ-stō′nē-ə, ĕ-stōn′yə). *Estonian* **Eesti** (ās′tĭ). Also **Es·tho·ni·a.** Officially, Estonian Soviet Socialist Republic. A constituent republic of the Soviet Union, occupying 17,400 square miles in northeastern Europe, along the Baltic Sea. Population, 1,357,000. Capital, Tallinn.

Es·to·ni·an (ĕs-tō′nē-ən) *adj.* Also **Es·tho·ni·an.** Pertaining to or characteristic of Estonia, its people, or their language. —*n.* Also **Es·tho·ni·an.** **1.** A native or inhabitant of Estonia. **2.** The Finno-Ugric language of Estonia.

es·top (ĕs-tŏp′) *tr.v.* **-topped, -topping, -tops.** **1.** *Law.* To prohibit, preclude, or impede by estoppel. **2.** *Archaic.* To stop up; plug up. [Middle English *estoppen*, from Old French *estoper, estouper*, from Late Latin *stuppāre*, to stop up. See **stop.**] —**es·top′page** (ĕs-tŏp′ĭj) *n.*

es·top·pel (ĕs-tŏp′əl) *n.* *Law.* A restraint on a person to prevent him from contradicting his own previous assertion. Also called "conclusion." [Old French *estoupail, estouppail*, from *estouper*, to ESTOP.]

es·to·vers (ĕs-tō′vərz) *pl.n.* Necessaries granted by law, as wood from a landlord to a tenant, alimony from a husband to a wife, or subsistence income from an estate to a beneficiary. [Middle English, from Norman French *estover*, from Old French *estoveir*, to be necessary, from Latin *est opus*, it is necessary : *est*, (it) is, from *esse*, to be (see **es-** in Appendix*) + *opus*, need, necessity (see **op-**[1] in Appendix*).]

es·tra·di·ol (ĕs′trə-dī′ōl) *n.* An estrogenic hormone, $C_{18}H_{24}O_2$, found in the follicle cells of ovaries and isolated commercially from sow ovaries or the urine of pregnant mares for use in treating estrogen deficiency. [ESTR(US) + DI- + -OL.]

es·tra·gon (ĕs′trə-gŏn; *French* ĕs-trà-gôN′) *n.* *French* Tarragon.

es·trange (ĕs-trānj′) *tr.v.* **-tranged, -tranging, -tranges.** **1.** To remove from an accustomed place or relation; put at a distance, especially a psychological distance. **2.** To alienate the affections of; make hostile or unsympathetic. [Old French *estranger, estrangier*, from Medieval Latin *extrāneāre*, from Latin *extrāneus*, STRANGE.] —**es·trange′ment** *n.* —**es·trang′er** *n.*

Synonyms: *estrange, alienate, disaffect.* These verbs refer to the disrupting of love, friendship, loyalty, or a similar bond. *Estrange* and *alienate* are often used with reference to two persons, typically a husband and wife or partners or coworkers, whose harmonious relationship has been replaced by hostility or indifference. *Estrange* generally implies separation. *Alienate* sometimes refers to a break caused by a third person. Both terms also can apply to disruption of a bond that existed between one or more persons and a group or institution. *Disaffect* usually refers to the disruption of loyalty or allegiance within the membership of a group.

es·tray (ĕ-strā′, ĭ-strā′) *n.* **1.** *Archaic.* A stray. **2.** *Law.* A stray domestic animal. —*intr.v.* **estrayed, -traying, -trays.** *Archaic.* To stray. [Norman French *estray*, from Old French *estraie*, stray, wandering, from *estraier*, to STRAY.]

Es·tre·cho de Ma·ga·lla·nes. The Spanish name for the Strait of **Magellan.**

Es·tre·ma·du·ra (ĕs′trə-mə-dŏŏr′ə; *Portuguese* ĕsh′trə-mə-dōō′rə). **1.** A province occupying 2,065 square miles in central Portugal. Population, 1,999,000. Capital, Lisbon. **2.** *Spanish* **Ex·tre·ma·du·ra** (ĕs′trā-mā-*thōō*′rā). A historic region of western Spain, once part of Roman Lusitania, along the Portuguese border.

es·tri·ol (ĕs′trē-ōl′) *n.* An estrogenic hormone, $C_{18}H_{24}O_3$, found in the ovaries of mammals, obtained commercially from the urine of pregnant animals, and used in treating estrogen deficiency. [ES(TRUS) + TRI- + -OL.]

es·tro·gen (ĕs′trə-jən) *n.* Also **oes·tro·gen** (ēs′-, ēs′-). Any of several steroid hormones produced chiefly by the ovary and responsible for promoting estrus and the development and maintenance of female secondary sex characteristics. Compare **androgen.** [ESTR(US) + -GEN.] —**es′tro·gen′ic** (-jĕn′ĭk) *adj.* —**es′tro·gen′i·cal·ly** *adv.*

es·trone (ĕs′trōn′) *n.* An estrogenic hormone, $C_{18}H_{22}O_2$, found in the mammalian ovary and isolated commercially from the urine of pregnant females for use in treating estrogen deficiency. Also called "theelin." [ESTR(US) + -ONE.]

es·trous (ĕs′trəs) *adj.* **1.** Of or pertaining to estrus. **2.** In heat. Said of an animal.

estrous cycle. The series of chemical and physiological changes in female mammals from one period of estrus to the next.

es·trus (ĕs′trəs) *n.* Also **oes·trus** (ēs′-, ĕs′trəs). A regularly recurrent period of ovulation and sexual excitement in female mammals other than humans. Also called "heat." [New Latin, from Latin *oestrus*, gadfly, frenzy, from Greek *oistros*. See **eis-**[1] in Appendix.*]

es·tu·a·rine (ĕs′chōō-ə-rīn, -rĭn′) *adj.* Of, pertaining to, or found in an estuary.

es·tu·ar·y (ĕs′chōō-ĕr′ē) *n.*, *pl.* **-ies.** **1.** The part of the wide lower course of a river where its current is met and influenced by the tides. **2.** An arm of the sea that extends inland to meet the mouth of a river. [Latin *aestuārium*, estuary, tidal channel, from *aestus*, heat, swell, surge, tide. See **aidh-** in Appendix.*] —**es′tu·ar′i·al** *adj.*

esu electrostatic unit.

e·su·ri·ent (ĭ-sŏŏr′ē-ənt) *adj.* Hungry; greedy: *"an esurient unprovided advocate: Danton"* (Carlyle). [Latin *ēsuriēns*, present participle of *ēsurīre*, to want food, to be hungry, desiderative of *edere* (past participle *ēsus*), to eat. See **ed-** in Appendix.*] —**e·su′ri·ence, e·su′ri·en·cy** *n.* —**e·su′ri·ent·ly** *adv.*

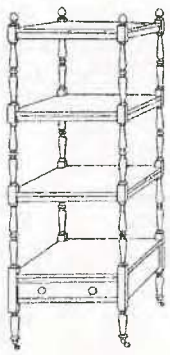
-et. Indicates smallness; for example, **falconet, spinneret.** [Middle English *-et*, from Old French *-et*, from Common Romance *-itta, -ētto* (both unattested).]

E.T. Eastern Time.

e·ta (ā′tə, ē′tə) *n.* The seventh letter in the Greek alphabet, written H, η. Transliterated in English as long *e*. See **alphabet.** [Late Latin *ēta*, from Greek, from a Phoenician source, akin to Hebrew *hēth*, HETH.]

e.t.a. estimated time of arrival.

é·ta·gère (ā′tä-zhâr′) *n.* Also **e·ta·gere.** A piece of furniture with open shelves for ornaments or bric-a-brac; whatnot. [French, from Old French *estagiere, estage*, floor of a building, position. See **stage.**]



étagère

et al. and others (Latin *et alii*).

etc. et cetera.

Usage: *Etc.* is principally appropriate to informal writing or to special areas such as technical reporting or business correspondence. It is not appropriate to formal writing in general.

et cet·er·a (ĕt sĕt′ər-ə, -sĕt′rə). Also **et-cet-er-a, et caet·er·a.** *Abbr.* **etc.** And other unspecified things of the same class; and so forth. [Latin, "and other (things)" : *et*, and (see **eti** in Appendix*) + *cētera*, the rest, from the neuter plural of *cēterus*, remaining (see **ko-** in Appendix*).]

et·cet·er·as (ĕt-sĕt′ər-əz, -sĕt′rəz) *pl.n.* A miscellany of extras; additional odds and ends.

etch (ĕch) *v.* **etched, etching, etches.** —*tr.* **1.** To wear away (metal or glass, for example) with or as if with acid. **2.** To make (a pattern) on a metal plate or other surface with acid. **3.** To impress or imprint clearly. —*intr.* To practice etching. [Dutch *etsen*, from German *ätzen*, to etch, to bite, to feed, from Old High German *ezzen*, to feed. See **ed-** in Appendix.*]

etch·ing (ĕch′ĭng) *n.* **1.** The art of preparing etched metal plates and printing designs and pictures with them. **2.** A design etched on a plate. **3.** An impression made from an etched plate.

e.t.d. estimated time of departure.

E·te·o·cles (ĭ-tē′ə-klēz′). *Greek Mythology.* A son of Oedipus and Jocasta.

e·ter·nal (ĭ-tûr′nəl) *adj.* **1.** Without beginning or end; existing outside of time: *God, the eternal Father.* **2.** Having a beginning but without interruption or end: *an eternal flame.* **3.** Unaffected by time; lasting; timeless. **4.** Seemingly endless; interminable. **5.** Of or relating to existence after death: *one's eternal reward.* —See Synonyms at **continual, infinite.** —*n.* Something eternal. —**the Eternal.** God. [Middle English, from Old French, from Late Latin *aeternālis*, from Latin *aeternus*, eternal. See **aiw-** in Appendix.*] —**e·ter′nal′i·ty, e·ter′nal·ness** *n.* —**e·ter′nal·ly** *adj.*

Eternal City. A name for Rome, Italy.

for sense 3) n. **1.** A troublesome, embarrassing, or ludicrous situation. **2.** *Archaic.* A specific state or condition. **3.** *Logic.* A state or classification of existence, a **category** *(see)*. [Middle English, from Late Latin *praedicāmentum* (translation of Greek *katēgoria*, category), something predicated, condition (especially an unpleasant one), from *praedicāre*, to proclaim; PREDICATE.] —**pre·dic′a·men′tal** *adj.* —**pre·dic′a·men′tal·ly** *adv.*

Synonyms: *predicament, plight, dilemma, quandary.* A *predicament* is a problematic situation seen in terms of a difficult decision and implies that one does not know what to do and is considering it rationally. A *plight* is a more serious pass, which may have been imposed on an individual with a course of action being less clear. *Dilemma* more abstractly denotes a problem which poses two alternatives, each of which must be carefully weighed. The term is sometimes loosely used of any problem. *Quandary*, somewhat more formal, suggests a complicated stalemate with numerous possibilities.

pred·i·cate (prĕd′ə-kāt′) *v.* **-cated, -cating, -cates.** —*tr.* **1.** To base or establish (a concept, statement, or action). Used with *on* or *upon*: *He predicates his argument on these facts.* See Usage note below. **2.** To state or affirm as an attribute or quality of something: *predicate perfectability of mankind.* **3.** To carry the connotation of; imply. **4.** *Logic.* To make (a term or expression) the predicate of a proposition. **5.** To proclaim; assert; declare. —*intr.* To make a statement or assertion. —*n.* (prĕd′ĭ-kĭt). *Abbr.* **pred.** **1.** *Grammar.* The part of a sentence or clause that expresses something about the subject. It regularly consists of a verb and may include objects, modifiers, or complements of the verb. The predicates of the following simple sentences are enclosed in brackets: *The house [is white.] The man [hit the dog.]* Compare **attributive. 2.** *Logic.* Whatever is stated about the subject of a proposition. —*adj.* (prĕd′ĭ-kĭt). **1.** *Grammar.* Of or belonging to the predicate of a sentence or clause. **2.** Predicated; stated. [Late Latin *praedicāre*, to proclaim, from Latin *prae*, in front of, in public + *dicāre*, to say (see **deik-** in Appendix*).] —**pred′i·ca′tive** *adj.* —**pred′i·ca′tive·ly** *adv.*

Usage: *Predicate* (transitive verb) is now employed widely in the sense of *base upon* or *found* as a means of indicating dependence: *Success is predicated on continuing effort.* Though this sense is disputed by some lexicographers, the foregoing typical example is acceptable to 62 per cent of the Usage Panel.

predicate adjective. *Grammar.* An adjective that follows certain verbs and describes the subject of the verb. In the sentence *The man is good*, the predicate adjective is *good.*

predicate nominative. *Grammar.* A noun, or a pronoun in the subject form, that follows certain verbs, and is identified with the subject of the verb. In the sentences *It is I* and *He became president*, the predicate nominatives are *I* and *president.*

pred·i·ca·tion (prĕd′ĭ-kā′shən) *n.* **1.** The act or procedure of predicating; especially, a logical assertion or affirmation. **2.** Something predicated. —**pred′i·ca′tion·al** *adj.*

pred·i·ca·to·ry (prĕd′ĭ-kə-tôr′ē, -tōr′ē) *adj.* Of, pertaining to, or characteristic of preaching or a preacher. [Late Latin *praedicātōrius*, from *praedicāre*, to proclaim; PREDICATE.]

pre·dict (pri-dĭkt′) *v.* **-dicted, -dicting, -dicts.** —*tr.* To state, tell about, or make known in advance, especially on the basis of special knowledge; foretell: *predict the weather.* —*intr.* To foretell what will happen; prophesy. —See Synonyms at **foretell.** [Latin *praedicere*, to foretell : *prae*, before + *dicere*, to tell, say (see **deik-** in Appendix*).] —**pre·dict′a·bil′i·ty** *n.* —**pre·dict′a·ble** *adj.* —**pre·dict′a·bly** *adv.*

pre·dic·tion (pri-dĭk′shən) *n.* **1.** The act of foretelling or predicting. **2.** Something foretold or predicted; prophecy. —**pre·dic′tive** *adj.* —**pre·dic′tive·ly** *adv.* —**pre·dic′tive·ness** *n.*

pre·dic·tor (pri-dĭk′tər) *n.* One that predicts.

pre·di·gest (prē′dī-jĕst′, -dĭ-jĕst′) *tr.v.* **-gested, -gesting, -gests.** To subject to partial digestion. —**pre′di·ges′tion** *n.*

pre·di·lec·tion (prĕd′ə-lĕk′shən, prē′də-) *n.* A preference, often formed as the result of personal leanings or disposition, rather than from objective knowledge. [French *prédilection*, from Medieval Latin *praedīligere*, to prefer : Latin *prae*, before + *dīligere*, to love, choose (see **dilligent**).]

pre·dis·pose (prē′dĭs-pōz′) *tr.v.* **-posed, -posing, -poses.** **1.** To make (someone) inclined to something in advance; put into a certain frame of mind for: *His good manners predispose people to like him.* **2.** To make susceptible or liable. **3.** *Archaic.* To settle or dispose of in advance. —**pre′dis-pos′al** *n.*

pre·dis·po·si·tion (prē′dĭs-pə-zĭsh′ən) *n.* The state of being predisposed; tendency or inclination.

pre·dom·i·nance (prĭ-dŏm′ə-nəns) *n.* Also **pre·dom·i·nan·cy** (-nən-sē). The state or quality of being predominant; preponderance.

pre·dom·i·nant (prĭ-dŏm′ə-nənt) *adj.* **1.** Having greatest ascendancy, importance, influence, authority, or force. **2.** Most common or conspicuous; prevalent. —See Synonyms at **dominant.** [Old French, from Medieval Latin *praedominâns*, present participle of *praedominārī*, PREDOMINATE.] —**pre·dom′i·nant·ly** *adv.*

pre·dom·i·nate (prĭ-dŏm′ə-nāt′) *v.* **-nated, -nating, -nates.** —*intr.* **1.** To be of greater power, importance, or quantity; be most important or outstanding. **2.** To have authority, power, or controlling influence; prevail. Often used with *over*. —*tr. Rare.* To dominate or prevail over. [Latin *praedominārī*, to subdue beforehand : *prae*, before + *dominārī*, to DOMINATE.] —**pre·dom′i·nate·ly** (-nĭt-lē), *adv.* —**pre·dom′i·nat′ing·ly** *adv.* —**pre·dom′i·na′tion** *n.* —**pre·dom′i·na′tor** (-nā′tər) *n.*

pree·mie (prē′mē) *n.* Also **pre·mie.** *Informal.* An infant born prematurely. [Short for PREMATURE.]

pre·em·i·nent (prē-ĕm′ə-nənt) *adj.* Also **pre·em·i·nent, pre·ĕm·i·nent.** Superior to or notable above all others; outstanding: *"Sublimity is the pre-eminent characteristic of the Paradise Lost."* (Emerson). See Synonyms at **dominant.** [Late Latin *praeēminēns*, from Latin, present participle of *praeēminēre*, to excel : *prae*, in front of + *ēminēre*, to stand out (see **eminent**).] —**pre·em′i·nence** *n.* —**pre·em′i·nent·ly** *adv.*

pre·empt (prē-ĕmpt′) *v.* **-empted, -empting, -empts.** Also **pre-empt, pre·ĕmpt.** —*tr.* **1.** To gain possession of by prior right or opportunity; especially, to settle on (public land) so as to obtain the right to buy before others. **2.** To appropriate, seize, or act for oneself before others. —*intr. Bridge.* To make a pre-emptive bid. [Back-formation from PRE-EMPTION.] —**pre-emp′tor** (-ĕmp′tər) *n.* —**pre-emp′to·ry** (-ĕmp′tə-rē) *adj.*

pre-emp·tion (prē-ĕmp′shən) *n.* Also **pre·emp·tion, pre·ĕmp·tion. 1. a.** The right to purchase something, especially government-owned land, before others. **b.** A purchase made when such a right is granted. **2.** Acquisition or appropriation of something beforehand. [Medieval Latin *praeemptiō*, from *praeemere*, to buy beforehand : *prae*, before + *emere*, to buy (see **em-** in Appendix*).]

pre-emp·tive (prē-ĕmp′tĭv) *adj.* Also **pre·emp·tive, pre·ĕmp·tive. 1.** Of, pertaining to, or characteristic of pre-emption. **2.** Having, or granted by, the right of pre-emption. **3.** *Bridge.* Designating or characteristic of a bid that is unnecessarily high, and is intended to prevent the opposing players from bidding. —**pre-emp′tive·ly** *adv.*

preen (prēn) *v.* **preened, preening, preens.** —*tr.* **1.** To smooth or clean (feathers) with the beak or bill. Used of a bird. **2.** To adorn or trim (oneself) carefully; primp. **3.** To take pride or satisfaction in (oneself). Used with *on*: *preening themselves on having won another victory.* —*intr.* To dress up; primp. [Middle English *preinen, proinen, prunen*, possibly from Old French *poroindre* (present stem *poroign-*), to anoint before : *por-*, from Latin *prō*, before + *oindre*, to anoint, from Latin *unguere* (see **ongw-** in Appendix*).] —**preen′er** *n.*

pre-es·tab·lish (prē′ə-stăb′lĭsh) *tr.v.* **-lished, -lishing, -lishes.** Also **pre·es·tab·lish, pre·ĕs·tab·lish.** To establish beforehand.

pre-ex·il·ic (prē′ĕg-zĭl′ĭk, -ĕk-sĭl′ĭk) *adj.* Also **pre-ex-il·i·an** (prē′ĕg-zĭl′ē-ən, -zĭl′yən, -ĕk-sĭl′ē-ən, -ĕk-sĭl′yən). Pertaining to the history of the Jewish people prior to their exile in Babylonia at the end of the sixth century B.C.

pre-ex·ist (prē′ĭg-zĭst′) *v.* **-isted, -isting, -ists.** Also **pre-ex·ist, pre·ĕx·ist.** —*intr.* To exist before. —*tr.* To exist before (something): *dinosaurs that pre-existed mammals.* —**pre-ex·ist′ence** *n.* —**pre-ex·ist′ent** *adj.*

pref. **1.** preface; prefatory. **2.** preference; preferred. **3.** prefix.

pre·fab (prē′făb′) *n.* A prefabricated part or structure.

pre·fab·ri·cate (prē-făb′rĭ-kāt′) *tr.v.* **-cated, -cating, -cates.** **1.** To construct or manufacture in advance. **2.** To construct in standard sections that can be easily shipped and assembled. —**pre-fab′ri·ca′tion** *n.* —**pre-fab′ri·ca′tor** (-kā′tər) *n.*

pref·ace (prĕf′ĭs) *n.* *Abbr.* **pref. 1. a.** A statement or essay, usually by the author, introducing a book and explaining its scope, intention, or background. **b.** The introductory section of a speech. **2.** Anything introductory or preliminary. **3.** *Usually capital P.* A thanksgiving prayer ending with the Sanctus and introducing the canon of the Roman Catholic Mass. —*tr.v.* **prefaced, -acing, -aces. 1.** To introduce by or provide with a preliminary statement or essay. **2.** To serve as an introduction to. [Middle English, from Old French, from Medieval Latin *prefātia*, alteration of Latin *praefātiō*, a saying beforehand, from *praefārī*, to say beforehand : *prae*, before + *fārī*, to speak (see **bhā-²** in Appendix*).] —**pref′ac·er** *n.*

pref·a·to·ry (prĕf′ə-tôr′ē, -tōr′ē) *adj.* Also **pref·a·to·ri·al** (prĕf′ə-tôr′ē-əl, -tōr′ē-əl). *Abbr.* **pref.** Of the nature of, or serving as, an introductory statement or essay; preliminary. [From Latin *prefātiō*, PREFACE.] —**pref′a·to′ri·ly** *adv.*

pre·fect (prē′fĕkt′) *n.* Also **pre·fect. 1.** Any of several high military or civil officials, as magistrates or administrators of ancient Rome. **2.** Any high administrative official; especially, the chief of police of Paris. **3.** The dean in a Jesuit school. **4.** A student officer, especially in a private school. [Middle English, from Old French, from Latin *praefectus*, overseer, chief, from the past participle of *praeficere*, to place at the head of : *prae*, before + *facere*, to do (see **dhē-¹** in Appendix*).]

prefect apostolic *pl.* **prefects apostolic.** A Roman Catholic priest with broad jurisdiction in a missionary territory.

pre·fec·ture (prē′fĕk′chər) *n.* **1.** The district, office, or authority of a prefect. **2.** The domicile or housing of a prefect. —**pre·fec′tur·al** *adj.*

pre·fer (prĭ-fûr′) *tr.v.* **-ferred, -ferring, -fers.** **1.** To select in preference to another or others; value more highly; like better. **2.** *Law.* To give priority or precedence to (a creditor). **3.** *Law.* To file, prosecute, or offer for consideration or resolution before a magistrate, court, or other legal person or body: *He preferred charges against her for theft.* **4.** To promote: *"Joey was now preferred from the stable to attend on the lady."* (Fielding). [Middle English *preferren*, from Old French *preferer*, from Latin *praeferre*, to hold or set before : *prae*, before + *ferre*, to bear (see **bher-¹** in Appendix*).] —**pre·fer′er** *n.*

Usage: *Prefer* is usually followed by *to* (never by *than*) when the object of *prefer* is something other than an infinitive: *We prefer reading to television.* When an infinitive is the object, the usual construction is *rather than* (not merely *than*) plus infinitive (sometimes without *to*): *We prefer to read rather than watch television* (but not *prefer to read than watch*). In stricter usage this can be expressed: *prefer to read instead of watching* or *would rather read than watch.*

pref·er·a·ble (prĕf′ər-ə-bəl) *adj.* More desirable or worthy;

EXHIBIT 5

INTRODUCTION TO THE THEORY OF NEURAL COMPUTATION

John Hertz
NORDITA

Anders Krogh
Niels Bohr Institute

Richard G. Palmer
Duke University and the Santa Fe Institute

Lecture Notes Volume I

SANTA FE INSTITUTE
STUDIES IN THE SCIENCES OF COMPLEXITY

Addison-Wesley Publishing Company
The Advanced Book Program
Redwood City, California • Menlo Park, California • Reading, Massachusetts
New York • Don Mills, Ontario • Wokingham, United Kingdom • Amsterdam
Bonn • Sydney • Singapore • Tokyo • Madrid • San Juan

Publisher: *Allan M. Wylde*
Production Manager: *Jan V. Benes*
Marketing Manager: *Laura Likely*

Director of Publications, Santa Fe Institute: *Ronda K. Butler-Villa*
Technical Assistant, Santa Fe Institute: *Della L. Ulibarri*

Hertz, John A.
Introduction to the theory of neural computation / John A. Hertz, Richard G. Palmer, Anders S. Krogh.
p. cm.—(Santa Fe Institute studies in the sciences of complexity. Lecture notes : v. 1) (Computation and neural systems series)
Includes index.
1. Neural computers. 2. Neural circuitry. I. Palmer, Richard G. II. Krogh, Anders S. III. Title. IV. Series. V. Series: Computation and neural systems series.
QA76.5.H475 1991 006.3—dc20 91-701
ISBN 0-201-50395-6.—ISBN 0-201-51560-1 (pbk.)

This volume was typeset using TEXtures on a Macintosh II computer. Camera-ready output from an Apple LaserWriter IINT Printer.

2345678910 - MA - 95 94 93 92 91

SIX
Multi-Layer Networks

The limitations of a simple perceptron do not apply to feed-forward networks with intermediate or "hidden" layers between the input and output layer. In fact, as we will see later, a network with just one hidden layer can represent any Boolean function (including for example XOR). Although the greater power of multi-layer networks was realized long ago, it was only recently shown how to make them *learn* a particular function, using "back-propagation" or other methods. This absence of a learning rule—together with the demonstration by Minsky and Papert [1969] that only linearly separable functions could be represented by simple perceptrons—led to a waning of interest in layered networks until recently.

Throughout this chapter, like the previous one, we consider only *feed-forward* networks. More general networks are discussed in the next chapter.

6.1 Back-Propagation

The back-propagation algorithm is central to much current work on learning in neural networks. It was invented independently several times, by Bryson and Ho [1969], Werbos [1974], Parker [1985] and Rumelhart et al. [1986a, b]. A closely related approach was proposed by Le Cun [1985]. The algorithm gives a prescription for changing the weights w_{pq} in any feed-forward network to learn a training set of input-output pairs $\{\xi_k^\mu, \zeta_i^\mu\}$. The basis is simply gradient descent, as described in Sections 5.4 (linear) and 5.5 (nonlinear) for a simple perceptron.

We consider first a two-layer network such as that illustrated by Fig. 6.1. Our notational conventions are shown in the figure; output units are denoted by O_i, hidden units by V_j, and input terminals by ξ_k. There are connections w_{jk} from the

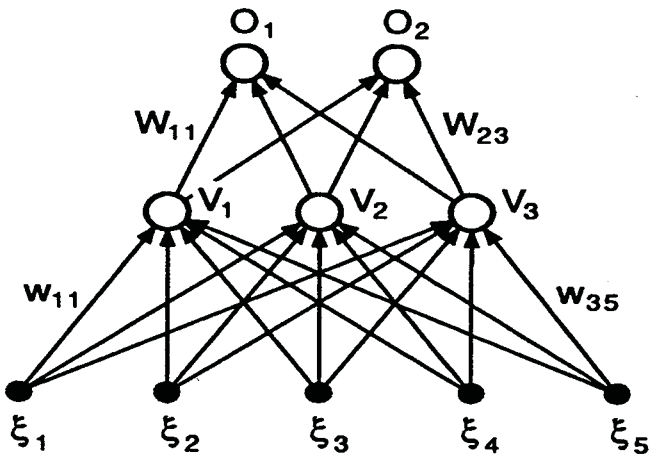


FIGURE 6.1 A two layer feed-forward network, showing the notation for units and weights.

inputs to the hidden units, and W_{ij} from the hidden units to the output units. Note that the index i always refers to an output unit, j to a hidden one, and k to an input terminal.

The inputs are always clamped to particular values. As in previous chapters, we label different patterns by a superscript μ, so input k is set to ξ_k^μ when pattern μ is being presented. The ξ_k^μ's can be binary (0/1, or ± 1) or continuous-valued. We use N for the number of *input* units and p, as before, for the number of input patterns ($\mu = 1, 2, \ldots, p$).

Given pattern μ, hidden unit j receives a net input

$$h_j^\mu = \sum_k w_{jk}\xi_k^\mu \tag{6.1}$$

and produces output

$$V_j^\mu = g(h_j^\mu) = g\Big(\sum_k w_{jk}\xi_k^\mu\Big). \tag{6.2}$$

Output unit i thus receives

$$h_i^\mu = \sum_j W_{ij}V_j^\mu = \sum_j W_{ij}g\Big(\sum_k w_{jk}\xi_k^\mu\Big) \tag{6.3}$$

and produces for the final output

$$O_i^\mu = g(h_i^\mu) = g\Big(\sum_j W_{ij}V_j^\mu\Big) = g\Big(\sum_j W_{ij}g\Big(\sum_k w_{jk}\xi_k^\mu\Big)\Big). \tag{6.4}$$

As in the previous chapter we have omitted the thresholds; they can be taken care of as usual by an extra input unit clamped to -1 and connected to all units in the network.

Our usual error measure or cost function

$$E[\mathbf{w}] = \frac{1}{2}\sum_{\mu i}[\zeta_i^\mu - O_i^\mu]^2 \tag{6.5}$$

now becomes

$$E[\mathbf{w}] = \frac{1}{2}\sum_{\mu i}\Big[\zeta_i^\mu - g\Big(\sum_j W_{ij}\, g\Big(\sum_k w_{jk}\xi_k^\mu\Big)\Big)\Big]^2 . \tag{6.6}$$

This is clearly a continuous differentiable function of every weight, so we can use a gradient descent algorithm to learn appropriate weights. In one sense this is all there is to back-propagation, but there is great practical importance in the form of the resulting update rules.

For the hidden-to-output connections the gradient descent rule gives

$$\begin{aligned}\Delta W_{ij} &= -\eta\frac{\partial E}{\partial W_{ij}} = \eta\sum_\mu[\zeta_i^\mu - O_i^\mu]g'(h_i^\mu)V_j^\mu \\ &= \eta\sum_\mu \delta_i^\mu V_j^\mu\end{aligned} \tag{6.7}$$

where we have defined

$$\delta_i^\mu = g'(h_i^\mu)[\zeta_i^\mu - O_i^\mu]. \tag{6.8}$$

The result is of course identical to that obtained earlier (equations (5.50) and (5.51)) for a single layer perceptron, with the output V_j^μ of the hidden units now playing the role of the perceptron input.

For the input-to-hidden connections Δw_{jk} we must differentiate with respect to the w_{jk}'s, which are more deeply embedded in (6.6). Using the chain rule, we obtain

$$\begin{aligned}\Delta w_{jk} &= -\eta\frac{\partial E}{\partial w_{jk}} = -\eta\sum_\mu \frac{\partial E}{\partial V_j^\mu}\frac{\partial V_j^\mu}{\partial w_{jk}} \\ &= \eta\sum_{\mu i}[\zeta_i^\mu - O_i^\mu]g'(h_i^\mu)W_{ij}g'(h_j^\mu)\xi_k^\mu \\ &= \eta\sum_{\mu i}\delta_i^\mu W_{ij}g'(h_j^\mu)\xi_k^\mu \\ &= \eta\sum_\mu \delta_j^\mu\xi_k^\mu\end{aligned} \tag{6.9}$$

with

$$\delta_j^\mu = g'(h_j^\mu)\sum_i W_{ij}\delta_i^\mu . \tag{6.10}$$

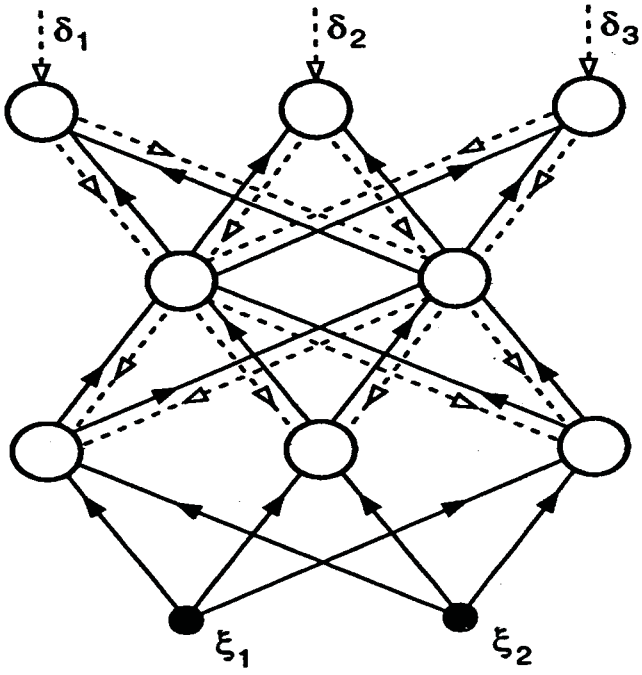


FIGURE 6.2 Back-propagation in a three-layer network. The solid lines show the forward propagation of signals and the dashed lines show the backward propagation of errors (δ's).

Note that (6.9) has the same form as (6.7), but with a different definition of the δ's. In general, with an arbitrary number of layers, the back-propagation update rule always has the form

$$\Delta w_{pq} = \eta \sum_{\text{patterns}} \delta_{\text{output}} \times V_{\text{input}} \tag{6.11}$$

where *output* and *input* refer to the two ends p and q of the connection concerned, and V stands for the appropriate input-end activation from a hidden unit or a real input. The meaning of δ depends on the layer concerned; for the last layer of connections it is given by (6.8), while for all other layers it is given by an equation like (6.10). It is easy to derive this generalized multi-layer result (6.11), simply by further application of the chain rule.

Equation (6.10) allows us to determine the δ for a given hidden unit V_j in terms of the δ's of the units O_i that it feeds. The coefficients are just the usual "forward" W_{ij}'s, but here they are propagating errors (δ's) backwards instead of signals forwards: hence the name **error back-propagation** or just **back-propagation**. We can therefore use the same network—or rather a bidirectional version of it—to compute both the output values and the δ's. Figure 6.2 illustrates this idea for a three-layer network.

Although we have written the update rules (6.7) and (6.9) as sums over all patterns μ, they are usually used incrementally: a pattern μ is presented at the input and then all weights are updated before the next pattern is considered. This clearly decreases the cost function (for small enough η) at each step, and lets successive steps adapt to the local gradient. If the patterns are chosen in random order it also

makes the path through weight-space *stochastic*, allowing wider exploration of the cost surface. The alternative **batch mode**—taking (6.7) and (6.9) literally and only updating after all patterns have been presented—requires additional local storage for each connection. The relative effectiveness of the two approaches depends on the problem, but the incremental approach seems superior in most cases, especially for very regular or redundant training sets.

The fact that the appropriate cost function derivatives can be calculated by back-propagating errors is clearly attractive. But it also has two important consequences:

- The update rule (6.11) is *local*. To compute the weight change for a given connection we only need quantities available (after back-propagation of the δ's) at the two ends of that connection. This makes the back-propagation rule appropriate for parallel computation. It may even have some indirect relevance for neurobiology.[1]
- The computational complexity is less than might have been expected. If we have n connections in all, computation of the cost function (6.6) takes of order n operations, so calculating n derivatives directly would take order n^2 operations. In contrast the back-propagation scheme lets us calculate *all* the derivatives in order n operations.

It is normal to use a sigmoid function for the activation function $g(h)$. The function clearly *must* be differentiable, and we normally want it to saturate at both extremes. Either a 0/1 or a ± 1 range can be used, with

$$g(h) = f_\beta(h) = \frac{1}{1+\exp(-2\beta h)} \tag{6.12}$$

and

$$g(h) = \tanh \beta h \tag{6.13}$$

respectively for the activation function. The steepness parameter β is often set to 1, or 1/2 for (6.12). As we noted in Chapter 5, the derivatives of these functions are readily expressed in terms of the functions themselves as $g'(h) = 2\beta g(1-g)$ for (6.12) and $g'(h) = \beta(1-g^2)$ for (6.13). Thus one often sees (6.8), for example, written as

$$\delta_i^\mu = O_i^\mu(1-O_i^\mu)(\zeta_i^\mu - O_i^\mu) \tag{6.14}$$

for 0/1 units with $\beta = 1/2$.

Because back-propagation is so important, we summarize the result in terms of a step-by-step procedure, taking one pattern μ at a time (i.e., incremental updates). We consider a network with M layers $m = 1, 2, \ldots, M$ and use V_i^m for the output

[1] Locality is necessary for biological implementation, but not sufficient. Bidirectional bifunctional connections are not biologically reasonable [Grossberg, 1987b], but can be avoided, allowing hypothetical neurophysiological implementations [Hecht-Nielsen, 1989]. Nevertheless back-propagation seems rather far-fetched as a biological learning mechanism [Crick, 1989].

of the ith unit in the mth layer. V_i^0 will be a synonym for ξ_i, the ith input. Note that superscript m's label layers, not patterns. We let w_{ij}^m mean the connection from V_j^{m-1} to V_i^m. Then the back-propagation procedure is:

1. Initialize the weights to small random values.
2. Choose a pattern ξ_k^μ and apply it to the input layer ($m = 0$) so that

$$V_k^0 = \xi_k^\mu \qquad \text{for all } k\,. \tag{6.15}$$

3. Propagate the signal forwards through the network using

$$V_i^m = g(h_i^m) = g\Big(\sum_j w_{ij}^m V_j^{m-1}\Big) \tag{6.16}$$

for each i and m until the final outputs V_i^M have all been calculated.

4. Compute the deltas for the output layer

$$\delta_i^M = g'(h_i^M)[\zeta_i^\mu - V_i^M] \tag{6.17}$$

by comparing the actual outputs V_i^M with the desired ones ζ_i^μ for the pattern μ being considered.

5. Compute the deltas for the preceding layers by propagating the errors backwards

$$\delta_i^{m-1} = g'(h_i^{m-1}) \sum_j w_{ji}^m \delta_j^m \tag{6.18}$$

for $m = M, M-1, \ldots, 2$ until a delta has been calculated for every unit.

6. Use

$$\Delta w_{ij}^m = \eta \delta_i^m V_j^{m-1} \tag{6.19}$$

to update all connections according to $w_{ij}^{\text{new}} = w_{ij}^{\text{old}} + \Delta w_{ij}$.

7. Go back to step 2 and repeat for the next pattern.

It is straightforward to generalize back-propagation to other kinds of networks where connections jump over one or more layers, such as the direct input-to-output connections in Fig. 6.5(b). This produces the same kind of error propagation scheme as long as the network is *feed-forward*, without any backward or lateral connections.

6.2 Variations on Back-Propagation

Back-propagation has been much studied in the past few years, and many extensions and modifications have been considered. The basic algorithm given above is exceedingly slow to converge in a multi-layer network, and many variations have

EXHIBIT 6

LEARNING FROM DATA

Concepts, Theory, and Methods

Vladimir Cherkassky

Filip Mulier

A WILEY-INTERSCIENCE PUBLICATION

JOHN WILEY & SONS, INC.

New York / Chichester / Weinheim / Brisbane / Singapore / Toronto

This book is printed on acid-free paper.

Published simultaneously in Canada.

Library of Congress Cataloging-in-Publication Data:

Cherkassky, Vladimir S.
Learning from data: concepts, theory, and methods / Vladimir Cherkassky and Filip Mulier.
p. cm.
"A Wiley-Interscience publication."
Includes bibliographical references and index.
ISBN 0-471-15493-8 (cloth: alk. paper)
1. Adaptive signal processing. 2. Machine learning. 3. Neural networks (Computer science) 4. Fuzzy systems. I. Mulier, Filip. II. Title.
TK5102.9.C475 1998
006.3'1--dc21 97-43019

Printed in the United States of America.

10 9 8 7 6 5 4 3 2 1

dependency). This view lends itself to mathematical treatment of learning (presented in Chapters 3 and 4), and hence is adopted throughout this book. However, in practice, due to complex and often informal nature of a priori knowledge, such specification of approximating functions may be difficult or impossible. Hence there may be a need to incorporate a priori knowledge into the learning method with an already given set of approximating functions. These issues are discussed in more detail in Section 2.3.

There is also an important distinction between two types of approximating functions: linear in parameters or nonlinear in parameters. Throughout this book learning (estimation) procedures using the former are also referred to as *linear*, whereas those using the latter are called *nonlinear*. We point out that the notion of linearity is with respect to parameters rather than input variables. For example, polynomial regression (2.2) is a linear method. Another example of a linear class of approximating functions (for regression) is the trigonometric expansion

$$f_m(x, \mathbf{v}_m, \mathbf{w}_m) = \sum_{j=1}^{m-1} (v_j \sin(jx) + w_j \cos(jx)) + w_0$$

On the other hand, multilayer networks of the form

$$f_m(\mathbf{x}, \mathbf{w}, V) = w_0 + \sum_{j=1}^{m} w_j g\left(v_{0j} + \sum_{i=1}^{d} x_i v_{ij}\right)$$

provide an example of nonlinear parameterization, since it depends nonlinearly on parameters V via nonlinear basis function g (usually taken as the so-called sigmoid activation function).

The distinction between linear and nonlinear methods is important in practice, since learning (estimation) of model parameters amounts to solving a linear or nonlinear optimization problem, respectively.

2.1.1 Role of the Learning Machine

The problem encountered by the learning machine is to select a function (from the set of functions it supports) that best approximates the system's response. The learning machine is limited to observing a finite number (n) examples in order to make this selection. This training data as produced by the generator and system will be independent and identically distributed (i.i.d.) according to the joint probability density function (pdf),

$$p(\mathbf{x}, y) = p(\mathbf{x})p(y \mid \mathbf{x}) \tag{2.5}$$

The finite sample (training data) from this distribution is denoted by

$$(\mathbf{x}_i, y_i), \qquad (i = 1, \ldots, n) \tag{2.6}$$

The quality of an approximation produced by the learning machine is measured by the loss $L(y, f(\mathbf{x}, \omega))$ or discrepancy between the output produced by the system and the learning machine for a given point $\mathbf{x}$. By convention, the loss takes on nonnegative values, so that large positive values correspond to poor approximation. The expected value of the loss is called the *risk functional*:

$$R(\omega) = \int L(y, f(\mathbf{x}, \omega)) p(\mathbf{x}, y)\, d\mathbf{x}\, dy \tag{2.7}$$

Learning is the process of estimating the function $f(\mathbf{x}, \omega_0)$, which minimizes the risk functional over the set of functions supported by the learning machine using only the training data ($p(\mathbf{x}, y)$ is not known). With finite data we cannot expect to find $f(\mathbf{x}, \omega_0)$ exactly, so we denote $f(\mathbf{x}, \omega^*)$ as the estimate of the optimal solution obtained with finite training data using some learning procedure. It is clear that any learning task (regression, classification, etc.) can be solved by minimizing (2.7) if the density $p(\mathbf{x}, y)$ is known. This means that density estimation is the most general (and hence most difficult) type of learning problem. The problem of learning (estimation) from finite data alone is inherently ill-posed. To obtain a useful (unique) solution, the learning process needs to incorporate a priori knowledge in addition to data. Let us assume that a priori knowledge is reflected in the set of approximating functions of a learning machine (as discussed earlier in this section). Then the next issue is: How should a learning machine use training data? The answer is given by the concept known as an *inductive principle*. An inductive principle is a general prescription for obtaining an estimate $f(\mathbf{x}, \omega^*)$ of the "true dependency" in the class of approximating functions, from the available (finite) training data. An inductive principle tells us *what* to do with the data, whereas the learning method specifies *how* to obtain an estimate. Hence a learning method (or algorithm) is a constructive implementation of an inductive principle for selecting an estimate $f(\mathbf{x}, \omega^*)$ from a particular set of functions $f(\mathbf{x}, \omega)$. For a given inductive principle there are many learning methods corresponding to a different set of functions of a learning machine. The distinction between inductive principles and learning methods is further discussed in Section 2.3.

2.1.2 Common Learning Tasks

The generic learning problem can be subdivided into four classes of common problems: classification, regression, density estimation, and clustering/vector quantization. For each of these problems, the nature of the loss function and

parameters) corresponding to the successive model estimates obtained during gradient-descent training. The solutions are penalized according to the number of gradient descent steps taken along this curve, namely the distance from the starting point (initial conditions) in the parameter space. This kind of penalization depends heavily on the particular optimization technique used, on the training data, and on the choice of (random) initial conditions. Hence it is difficult to control and interpret such "penalization" via early stopping rules (Friedman, 1994).

Structural Risk Minimization (SRM) Under SRM, approximating functions of a learning machine are ordered according to their complexity, forming a *nested structure*:

$$S_0 \subset S_1 \subset S_2 \subset \ldots \tag{2.54}$$

For example, in the class of polynomial approximating functions, the elements of a structure are polynomials of a given degree. Condition (2.54) is satisfied, since polynomials of degree m are a subset of polynomials of degree $(m + 1)$. The goal of learning is to choose an optimal element of a structure (i.e., polynomial degree) and estimate its coefficients from a given training sample. For approximating functions *linear* in parameters such as polynomials, the complexity is given by the number of free parameters. For functions nonlinear in parameters, the complexity is defined as VC-dimension (see Chapter 4). The optimal choice of model complexity provides the minimum of the expected risk. Statistical learning theory (Vapnik, 1995) provides analytic upper-bound estimates for expected risk. These estimates are used for model selection, namely choosing an optimal element of a structure under the SRM inductive principle.

Bayesian Inference The Bayesian type of inference uses additional a priori information about approximating functions in order to obtain a unique predictive model from finite data. This knowledge is in the form of the so-called prior probability distribution, which is the probability of any function (from the set approximating functions) being the true (unknown) function. Note that the prior distribution usually reflects *subjective* degree of belief (in the sense described in Section 1.4). This adds subjectivity to the design of a learning machine, since the final model depends largely on a good choice of priors. Moreover the very notion that the prior distribution adequately captures prior knowledge may not be acceptable in many situations, namely where we need to estimate a constant (but unknown) parameter. On the other hand, the Bayesian approach provides an effective way of encoding prior knowledge, and it can be a powerful tool when used by experts.

Bayesian inference is based on the classical Bayes formula for updating

prior probabilities using the evidence provided by the data:

$$P[\text{model} \mid \text{data}] = \frac{P[\text{data} \mid \text{model}]P[\text{model}]}{P[\text{data}]} \tag{2.55}$$

where

P[model] is *prior* probability (before the data are observed),
P[data] is the probability of observing training data,
P[model | data] is *a posterior* probability of a model given the data,
P[data | model] is the probability that the data are generated by a model, also known as the *likelihood*.

Let us consider the general case of (parametric) density estimation where the class of density functions supported by the learning machine is a parametric set, namely $f(\mathbf{x}, \mathbf{w})$, $\mathbf{w} \in \Omega$ is a set of densities where $\mathbf{w}$ is an m-dimensional vector of "free" parameters (m is fixed). It is also assumed that the unknown density $f(\mathbf{x}, \mathbf{w}_0)$ belongs to this class. Given a set of i.i.d. training data $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$, the probability of seeing this particular data set as a function of $\mathbf{w}$ is

$$P[\text{data} \mid \text{model}] = P(\mathbf{W} \mid \mathbf{w}) = \prod_{i=1}^{n} f(\mathbf{x}_i, \mathbf{w}) \tag{2.56}$$

(recall that choosing the model, i.e., parameter $\mathbf{w}^*$, maximizing likelihood $P(X \mid \mathbf{w})$ amounts to ML inference discussed in Section 2.2.2). The a priori density function

$$P[\text{model}] = p(\mathbf{w}) \tag{2.57}$$

gives the probability of any (implementable) density $f(\mathbf{x}, \mathbf{w})$, $\mathbf{w} \in \Omega$ being the true one. Then Bayes formula gives

$$p(\mathbf{w} \mid \mathbf{X}) = \frac{P(\mathbf{X} \mid \mathbf{w})p(\mathbf{w})}{P(\mathbf{X})} \tag{2.58}$$

Usually the prior distribution is taken rather broadly, reflecting general uncertainty about "correct" parameter values. Having observed the data, this prior distribution is converted into posterior distribution according to

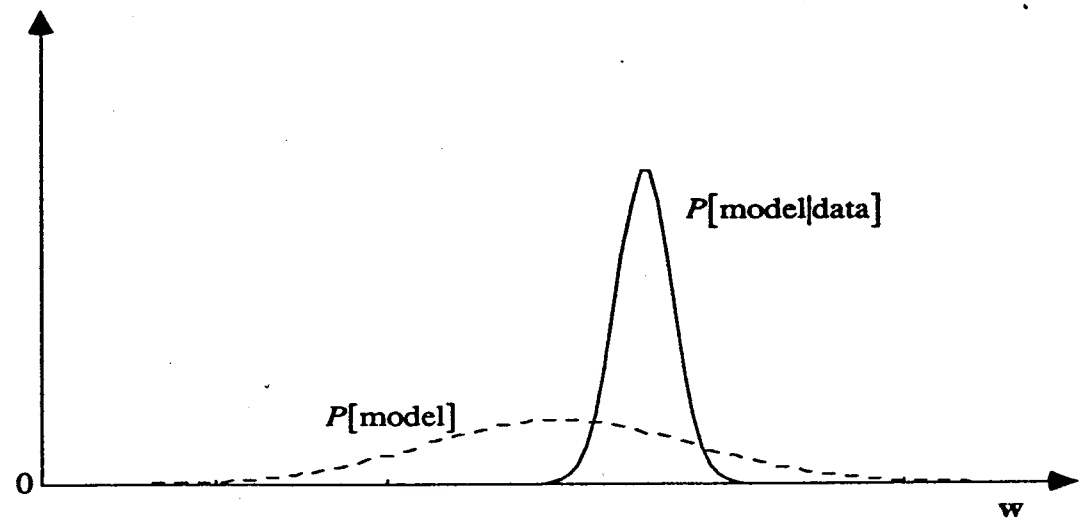


Figure 2.6 After observing the data, the wide prior distribution is converted into the more narrow posterior distribution using Bayes rule.

Bayes formula. This posterior distribution will be more narrow, reflecting the fact that it is consistent with the observed data; see Fig. 2.6.

There are two distinct ways to use Bayes formula for obtaining an estimate of unknown p.d.f. The true Bayesian approach is to average over all possible models (implementable by a learning machine), which gives the following p.d.f. estimate:

$$\Theta(\mathbf{x} \mid \mathbf{X}) = \int f(\mathbf{x}, \mathbf{w}) p(\mathbf{w} \mid \mathbf{X}) \, d\mathbf{w} \tag{2.59}$$

where $p(\mathbf{w} \mid \mathbf{X})$ is given by the Bayes formula (2.58). Equation (2.59) provides an example of an important technique in Bayesian inference called *marginalization*, which involves integrating out redundant variables, such as parameters $\mathbf{w}$. The estimator $\Theta(\mathbf{x} \mid \mathbf{X})$ has many attractive properties (Bishop, 1995). In particular, the final model is a weighted sum of all possible predictive models, with weights given by the evidence (or posterior probability) that each model is correct. However, multidimensional integration (due to the large number of parameters $\mathbf{w}$) presents a challenging problem. Standard numerical integration is impossible, whereas analytic evaluation may be possible only under restrictive assumptions when the posterior density has the same form as a prior (typically assumed to be gaussian) and $f(\mathbf{x}, \mathbf{w})$ is linear in parameters $\mathbf{w}$. When gaussian assumptions do not hold, various forms of random sampling, also known as Monte Carlo methods, have been proposed to evaluate integrals (2.59) directly (Bishop, 1995).

Another (simpler) way to implement the Bayesian approach is to choose an estimate $f(\mathbf{x}, \mathbf{w}^*)$ maximizing posterior probability $p(\mathbf{w} \mid \mathbf{X})$. This is known

as the maximum a posterior probability (MAP) estimate. This is mathematically equivalent to the penalization formulation, as explained next.

Let us consider regression formulation of the learning problem, namely the training data $(\mathbf{x}_i, y_i)$ generated according to

$$y = f(\mathbf{x}, \mathbf{w}_0) + \epsilon \tag{2.60}$$

To estimate an unknown function from the training data $\mathbf{Z} = [\mathbf{X}, \mathbf{y}]$, where $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n]$ and $\mathbf{y} = [y_1, \ldots, y_n]$, we need to assume that the set of parametric functions (of a learning machine) $f(\mathbf{x}, \mathbf{w})$ contains the true one. In addition under the Bayesian approach we need to know a priori density $p(\mathbf{w})$ specifying the probability of any admissible $f(\mathbf{x}, \mathbf{w})$ to be the true one. The Bayes formula gives a posterior probability that parameter $\mathbf{w}$ specifies the unknown function

$$p(\mathbf{w} \mid \mathbf{Z}) = \frac{P(\mathbf{Z} \mid \mathbf{w})p(\mathbf{w})}{P(\mathbf{Z})} \tag{2.61}$$

where the probability that the training data are generated by the model $f(\mathbf{x}, \mathbf{w})$ is

$$P(\mathbf{Z} \mid \mathbf{w}) = \prod_{i=1}^{n} p(\mathbf{x}_i, y_i) = P(\mathbf{X}) \prod_{i=1}^{n} p(y_i - f(\mathbf{x}_i, \mathbf{w})) \tag{2.62}$$

Substituting (2.62) into (2.61), taking the logarithm of both sides, and discarding terms that do not depend on parameters $\mathbf{w}$ gives an equivalent functional for MAP estimation:

$$R_{\text{map}}(\mathbf{w}) = \sum \ln p(y_i - f(\mathbf{x}_i, \mathbf{w})) + \ln p(\mathbf{w}) \tag{2.63}$$

The value of $\mathbf{w}^*$ maximizing this functional gives maximum a posterior probability. Further assume that error has gaussian distribution:

$$\epsilon_i = y_i - f(\mathbf{x}_i, \mathbf{w}_0) \sim N(0, \sigma^2) \tag{2.64}$$

then

$$\ln p(y_i - f(\mathbf{x}_i, \mathbf{w})) = (y_i - f(\mathbf{x}_i, \mathbf{w}))^2 \tag{2.65}$$

so

$$R_{\text{map}}(\mathbf{w}) = \frac{1}{n} \sum (y_i - f(\mathbf{x}_i, \mathbf{w}))^2 + \frac{2\sigma^2}{n} \ln p(\mathbf{w}) \tag{2.66}$$

Thus MAP formulation is equivalent to penalization formulation (2.53) with an explicit form of regularization parameter (reflecting the knowledge

of noise variance). If the noise variance is not known, it can be estimated (from data), and this is equivalent to estimating regularization parameter (using resampling methods). Hence penalization formulation has a natural Bayesian interpretation, so the choice of a penalty term corresponds to a priori information (distribution) about the target function, and the choice of the regularization parameter reflects knowledge (or estimate) of the amount of noise (i.e., its variance). For very large noise, the prior knowledge completely specifies the MAP solution; for zero noise, the solution is completely determined by the data (interpolation problem).

Choosing the value of regularization parameter is equivalent to finding a "good" prior. There has been some work done to tailor priors to the data, namely using the so-called type II maximum likelihood or MLII techniques (Berger, 1985). However, tailoring priors to the data contradicts the original notion of data-independent prior knowledge. On one hand, the prior distribution is (by definition) independent of the data (i.e., the number of samples). On the other hand, the prior effectively controls model complexity, as is evident from the connection between MAP and penalization formulation. The optimal prior is equivalent to the choice of the regularization parameter, which clearly depends on the sample size as in (2.66).

Although the penalization inductive principle can, in some cases, be interpreted in terms of a Bayesian formulation, penalization and Bayesian methods have a different motivation. The Bayesian methodology is used to encode a priori knowledge about multiple, general, user-defined characteristics of the target function. The goal of penalization is to perform complexity control by encoding a priori knowledge about function smoothness in terms of a penalty functional. Bayesian model selection tends to penalize more complex models in choosing the model with the largest evidence, but this does not guarantee the best generalization performance (or minimum prediction risk). On the other hand, formulations provided by penalization framework and SRM are based on the explicit minimization of the prediction risk.

Bayesian approach can be also used to compare several (potential) classes of approximating functions. For example, let us consider two (parametric) models

$$M_1 = f_1(\mathbf{x}, \mathbf{w}_1) \quad \text{and} \quad M_2 = f_2(\mathbf{x}, \mathbf{w}_2)$$

Say these models are feedforward networks with a different number of hidden units. Our problem is to choose the best model to describe a given (training) data set $\mathbf{Z}$. Using Bayes formula (2.55), we can estimate relative plausibilities of the two models using the so-called Bayes factor:

$$\frac{P(M_1 \mid \mathbf{Z})}{P(M_2 \mid \mathbf{Z})} = \frac{P(\mathbf{Z} \mid M_1)P(M_1)}{P(\mathbf{Z} \mid M_2)P(M_2)} \tag{2.67}$$

where $P(M_1)$ and $P(M_2)$ are the prior probabilities assigned to each model (usually assumed to be the same) and $P(\mathbf{Z} \mid M_i)$ is the "evidence" of the model M_i calculated as

$$P(\mathbf{Z} \mid M_i) = \int P(\mathbf{Z}, \mathbf{w}_i \mid M_i)\, d\mathbf{w}_i = \int P(\mathbf{Z} \mid \mathbf{w}_i, M_i) p(\mathbf{w}_i \mid M_i)\, d\mathbf{w}_i \quad (2.68)$$

Thus Bayesian approach enables, in principle, model selection without resorting to data-driven (resampling) techniques. However, the difficulty of multidimensional integration (2.68) limits practical applicability of this approach.

Minimum Description Length (MDL) The MDL principle is based on the information-theoretic analysis of the randomness concept. In contrast to all other inductive principles which use statistical distributions to describe an unknown model, this approach regards *models as codes*, that is, as encodings of the training data. The main idea is that any data set can be appropriately encoded, and its *code length* represents an inherent property of the data which is directly related to the generalization capability of the model (i.e., code).

Kolmogorov (1965) introduced the notion of *algorithmic complexity* for characterization of randomness of a data set. He defined the algorithmic complexity of a data set to be the shortest binary code describing this data. Further the randomness of a data set can be related to the length of the binary code; that is, the data samples are random if they cannot be compressed significantly. Rissanen (1978) proposed using Kolmogorov's characterization of randomness as tool for inductive inference; this is known as the MDL principle.

To illustrate the MDL inductive principle, we consider the training data set

$$(\mathbf{x}_i, y_i), \qquad i = 1, \ldots, n$$

where samples $(\mathbf{x}_i, y_i)$ are drawn randomly and independently from some (unknown) distribution. Let us further assume that training data corresponds to classification problem, where the class label $y = \{0, 1\}$ and $\mathbf{x}$ is d-dimensional feature vector. The problem of estimating dependency between $\mathbf{x}$ and y can be formulated under MDL inductive principle as follows: Given a data object $\mathbf{X} = (\mathbf{x}_1, \ldots, \mathbf{x}_n)$, is a binary string $y_1, \ldots, y_n$ random?

The binary string $\mathbf{y} = (y_1, \ldots, y_n)$ can be encoded using n bits. However, if there are systematic dependency in the data captured by the model $y = f(\mathbf{x})$, we can encode the output string $\mathbf{y}$ by a possibly shorter code that consists of two parts: the model having code length L(model) and the error term specifying how the actual data differs from the model predictions, with

following stochastic approximation procedure updates parameter values during each presentation of kth training sample:

Step 1. Forward pass computations

$$z_j(k) = g_j(\mathbf{x}(k)), \qquad j = 1, \ldots, m \tag{5.8}$$

$$\hat{y}(k) = \sum_{j=1}^{m} w_j(k) z_j(k) \tag{5.9}$$

Step 2. Backward pass computations

$$\delta(k) = \hat{y}(k) - y(k) \tag{5.10}$$

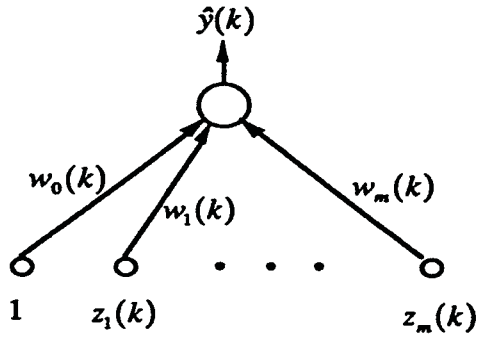
$$\mathbf{w}_j(k+1) = \mathbf{w}_j(k) - \gamma_k \delta(k) z_j(k), \qquad j = 1, \ldots, m \tag{5.11}$$

where the learning rate γ_k is a small positive number (usually) decreasing with k as prescribed by stochastic approximation theory, that is, conditions (2.46). Note that the factor 2 in (5.7) can be absorbed in the learning rate. In the forward pass, the output of the approximating function is computed, storing some intermediate results. In the backward pass, the error term (5.7) for the presented sample is calculated and used to adjust the parameters. The error term is often called "delta" in signal-processing and neural network literature, and the parameter updating scheme (5.11) is known as the delta rule (Widrow and Hoff, 1960). The delta rule effectively implements least-mean-squares (LMS) minimization in an online (or flow-through) fashion, updating parameters with every training sample.

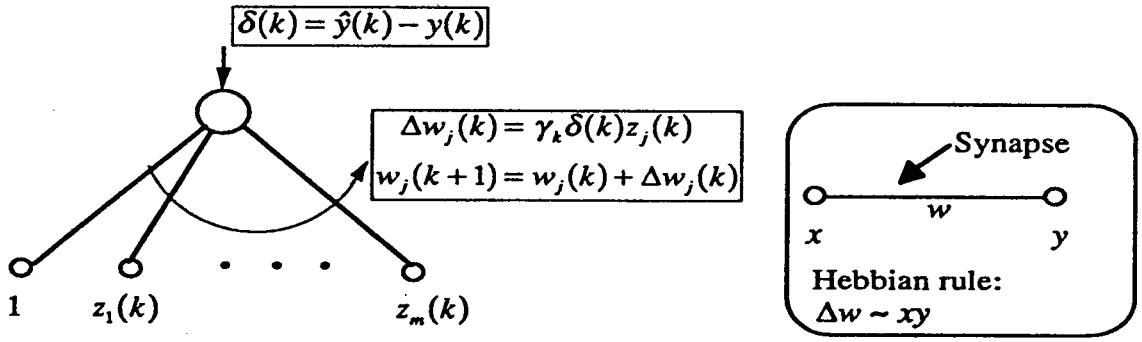
Equations (5.10) and (5.11) have a convenient "neural network" interpretation where parameters correspond to the (adjustable) "synaptic weights" of a neural network, and input/output variables are represented as network units or "neurons" (see Fig. 5.1). Then according to (5.11) the change in connection strength (between a pair of input-output units) is proportional to the error (observed by the output unit) and to activation of the input unit. This corresponds to the well-known Hebbian rule describing (qualitatively) operation of the biological neurons (see Fig. 5.1).

5.1.2 Backpropagation Training of MLP Networks

As an example of stochastic approximation strategy for nonlinear approximating functions, we consider next a popular optimization (or training) method for MLP networks called *backpropagation* (Werbos, 1974, 1994). Consider a learning machine implementing the ERM inductive principle with



(a) Forward pass



(b) Backward pass

Figure 5.1 Neural network interpretation of the delta rule.

L_2 loss function and a set of approximating functions given by

$$f(\mathbf{x}, \mathbf{w}, \mathbf{V}) = w_0 + \sum_{j=1}^{m} w_j g\left(v_{0j} + \sum_{i=1}^{d} x_i v_{ij}\right) \tag{5.12}$$

where the function g is a differentiable monotonically increasing function called the activation function. Parameterization (5.12) is known as a multilayer perceptron (MLP) with a single layer of hidden units, where a hidden unit corresponds to the basis function in (5.12). Note that in contrast to (5.5), this set of functions is nonlinear in the parameters $\mathbf{V}$. However, the gradient descent approach can still be applied. The risk functional is

$$R_{\text{emp}} = \sum_{i=1}^{n} (f(\mathbf{x}_i, \mathbf{w}, \mathbf{V}) - y_i)^2 \tag{5.13}$$

The stochastic approximation procedure for minimizing this risk with respect to the parameters $\mathbf{V}$ and $\mathbf{w}$ is

$$\mathbf{V}(k+1) = \mathbf{V}(k) - \gamma_k \operatorname{grad}_{\mathbf{v}} L(\mathbf{x}(k), y(k), \mathbf{V}(k), \mathbf{w}(k)) \tag{5.14}$$

$$\mathbf{w}(k+1) = \mathbf{w}(k) - \gamma_k \operatorname{grad}_{\mathbf{w}} L(\mathbf{x}(k), y(k), \mathbf{V}(k), \mathbf{w}(k)), \qquad k = 1, \ldots, n, \ldots \tag{5.15}$$

where $\mathbf{x}(k)$ and $y(k)$ are the kth training sample, presented at iteration step k. The loss L is

$$L(\mathbf{x}(k), y(k), \mathbf{V}(k), \mathbf{w}(k)) = \tfrac{1}{2}(f(\mathbf{x}, \mathbf{w}, \mathbf{V}) - y)^2 \tag{5.16}$$

for a given data point $(\mathbf{x}, y)$ with respect to the parameters $\mathbf{w}$ and $\mathbf{V}$. (The constant $\frac{1}{2}$ is included to streamline gradient calculations). The gradient of (5.16) can be computed via the chain rule of derivatives if the approximating function (5.12) is decomposed as

$$a_j = \sum_{i=0}^{d} x_i v_{ij}, \qquad j = 1, \ldots, m \tag{5.17}$$

$$z_j = g(a_j), \qquad j = 1, \ldots, m \tag{5.18}$$

$$z_0 = 1$$

$$\hat{y} = \sum_{j=0}^{m} w_j z_j \tag{5.19}$$

To simplify notation, we drop the iteration step k and consider the gradient calculation/parameter update for one sample at a time; the zeroth-order terms w_0 and v_{0j} have been incorporated into the summations (by setting $x_0 = 1$). Based on the chain rule, the relevant gradients are

$$\frac{\partial L}{\partial v_{ij}} = \frac{\partial L}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial a_j} \frac{\partial a_j}{\partial v_{ij}} \tag{5.20}$$

$$\frac{\partial L}{\partial w_j} = \frac{\partial L}{\partial \hat{y}} \frac{\partial \hat{y}}{\partial w_j} \tag{5.21}$$

Each of these partial derivatives can be calculated based on (5.16) through (5.19). From (5.16) we can calculate

$$\frac{\partial L}{\partial \hat{y}} = \hat{y} - y \tag{5.22}$$

From (5.18) and (5.19) we determine

$$\frac{\partial \hat{y}}{\partial a_j} = g'(a_j) w_j \tag{5.23}$$

From (5.17) we get

$$\frac{\partial a_j}{\partial v_{ij}} = x_i \tag{5.24}$$

From (5.19) we find

$$\frac{\partial \hat{y}}{\partial w_j} = z_j \tag{5.25}$$

Plugging these partial derivatives into (5.20) and (5.21) gives the gradient equations:

$$\frac{\partial L}{\partial v_{ij}} = (\hat{y} - y)g'(a_j)w_j x_i \tag{5.26}$$

$$\frac{\partial L}{\partial w_j} = (\hat{y} - y)z_j \tag{5.27}$$

With these gradients and the stochastic approximation updating equations, it is now possible to construct a computational procedure to minimize the empirical risk. Starting with an initial guess for values $\mathbf{w}(0)$ and $\mathbf{V}(0)$, the stochastic approximation procedure for parameter (weight) updating upon presentation of a sample $(\mathbf{x}(k), y(k))$ at iteration step k with learning rate γ_k is as follows:

Step 1. Forward pass computations

"Hidden layer"

$$a_j(k) = \sum_{i=0}^{d} x_i(k)v_{ij}(k), \qquad j = 1, \ldots, m \tag{5.28}$$

$$z_j(k) = g(a_j(k)), \qquad j = 1, \ldots, m$$

$$z_0(k) = 1 \tag{5.29}$$

"Output layer"

$$\hat{y}(k) = \sum_{j=0}^{m} w_j(k)z_j(k) \tag{5.30}$$

Step 2. Backward pass computations

"Output layer"

$$\delta_0(k) = \hat{y}(k) - y(k) \tag{5.31}$$

$$w_j(k+1) = w_j(k) - \gamma_k \delta_0(k) z_j(k), \qquad j = 0, \ldots, m \tag{5.32}$$

"Hidden layer"

$$\delta_{1j}(k) = \delta_0(k) g'(a_j(k)) w_j(k+1), \qquad j = 0, \ldots, m \tag{5.33}$$

$$v_{ij}(k+1) = v_{ij}(k) - \gamma_k \delta_{1j}(k) x_i(k), \qquad i = 0, \ldots, d, \; j = 0, \ldots, m \tag{5.34}$$

In the forward pass, the output of the approximating function is computed, storing some intermediate results that will be required in the next step. In the backward pass, the error difference for the presented sample is first calculated and used to adjust the parameters in the output layer. Via the chain rule it is possible to relate (or propagate) the error at the output back to an error at each of the internal nodes a_j, $j = 1, \ldots, m$. This is called *error backpropagation* because it can be conveniently represented in graphical form as a propagation of the (weighted) error signals from the output layer back to the input layer (see Fig. 5.2). Note that the updating steps for the output layer (5.31), (5.32) are identical to those for the linear parameter estimation (5.10), (5.11). Also the updating rule for the hidden layer is similar to the linear case, except for the delta term (5.33). Hence backpropagation update rules (5.33), (5.34) are sometimes called the "generalized delta rule" in the neural network literature. The parameter update algorithm presented in this section assumes a stochastic approximation setting when the number of training samples is large (infinite). In practice, the sample size is finite, and asymptotic conditions of stochastic approximation are (approximately) satisfied by the repeated presentation of the finite training sample to the training algorithm. This is known as recycling, and the number of such repeated presentations of the complete training set is called the number of cycles (or epochs). Detailed discussion on these and other implementation details of backpropagation (initialization of parameter values, choice of the learning rate schedule, etc.) will be presented in Chapter 7.

The equations given above are for a single hidden layer, single (linear) output unit network, corresponding to regression problems with a single output variable. Obvious generalizations include networks with several output units and networks with several hidden layers (of nonlinear units). The above backpropagation algorithm can be readily extended to these types of networks. For example, if additional "layers" are added to the approximating function, then errors are "backpropagated" from layer to layer by repeated application of equations (5.33) and (5.34).

Note that the backpropagation training is not limited to the squared loss error function. Other loss functions can be used as long as partial derivatives of the risk functional (with respect to parameters) can be calculated via the chain rule.

EXHIBIT 7

FULLY REDACTED

EXHIBIT 8

Machine learning

From Wikipedia, the free encyclopedia

Machine learning is a scientific discipline that is concerned with the design and development of algorithms that allow computers to evolve behaviors based on empirical data, such as from sensor data or databases. A learner can take advantage of examples (data) to capture characteristics of interest of their unknown underlying probability distribution. Data can be seen as examples that illustrate relations between observed variables. A major focus of machine learning research is to automatically learn to recognize complex patterns and make intelligent decisions based on data; the difficulty lies in the fact that the set of all possible behaviors given all possible inputs is too large to be covered by the set of observed examples (training data). Hence the learner must generalize from the given examples, so as to be able to produce a useful output in new cases. Artificial intelligence is a closely related field, as are probability theory and statistics, data mining, pattern recognition, adaptive control, computational neuroscience and theoretical computer science.

Contents

Definition

> A computer program is said to **learn** from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.[1]

Generalization

The core objective of a learner is to generalize from its experience.[2] The training examples from its experience come from some generally unknown probability distribution and the learner has to extract from them something more general, something about that distribution, that allows it to produce useful answers in new cases.

Human interaction

Some machine learning systems attempt to eliminate the need for human intuition in data analysis, while others adopt a collaborative approach between human and machine. Human intuition cannot, however, be entirely eliminated, since the system's designer must specify how the data is to be represented and what mechanisms will be used to search for a characterization of the data.

Algorithm types

Machine learning algorithms are organized into a taxonomy, based on the desired outcome of the algorithm.

- **Supervised learning** generates a function that maps inputs to desired outputs. For example, in a classification problem, the learner approximates a function mapping a vector into classes by looking at input-output examples of the function.
- **Unsupervised learning** models a set of inputs, like clustering.
- **Semi-supervised learning** combines both labeled and unlabeled examples to generate an appropriate function or classifier.
- **Reinforcement learning** learns how to act given an observation of the world. Every action has some impact in the environment, and the environment provides feedback in the form of rewards that guides the learning algorithm.
- **Transduction** tries to predict new outputs based on training inputs, training outputs, and test inputs.
- **Learning to learn** learns its own inductive bias based on previous experience.

Theory

Main article: Computational learning theory

The computational analysis of machine learning algorithms and their performance is a branch of theoretical computer science known as computational learning theory. Because training sets are finite and the future is uncertain, learning theory usually does not yield absolute guarantees of the performance of algorithms. Instead, probabilistic bounds on the performance are quite common.

In addition to performance bounds, computational learning theorists study the time complexity and feasibility of learning. In computational learning theory, a computation is considered feasible if it can be done in polynomial time. There are two kinds of time complexity results. Positive results show that a certain class of functions can be learned in polynomial time. Negative results show that certain classes cannot be learned in polynomial time.

There are many similarities between machine learning theory and statistics, although they use different terms.

Approaches

Main article: List of machine learning algorithms

Decision tree learning

Main article: Decision tree learning

Decision tree learning uses a decision tree as a predictive model which maps observations about an item to conclusions about the item's target value.

Association rule learning

Main article: Association rule learning

Association rule learning is a method for discovering interesting relations between variables in large databases.

Artificial neural networks

Main article: Artificial neural network

An artificial neural network (ANN), usually called "neural network" (NN), is a mathematical model or computational model that tries to simulate the structure and/or functional aspects of biological neural networks. It consists of an interconnected group of artificial neurons and processes information using a connectionist approach to computation. Modern neural networks are non-linear statistical data modeling tools. They are usually used to model complex relationships between inputs and outputs or to find patterns in data.

Genetic programming

Main articles: Genetic programming and Evolutionary computation

Genetic programming (GP) is an evolutionary algorithm-based methodology inspired by biological evolution to find computer programs that perform a user-defined task. It is a specialization of genetic algorithms (GA) where each individual is a computer program. It is a machine learning technique used to optimize a population of computer programs according to a fitness landscape determined by a program's ability to perform a given computational task.

Inductive logic programming

Main article: Inductive logic programming

Inductive logic programming (ILP) is an approach to rule learning using logic programming as a uniform representation for examples, background knowledge, and hypotheses. Given an encoding of the known background knowledge and a set of examples represented as a logical database of facts, an ILP system will derive a hypothesized logic program which entails all the positive and none of the negative examples.

Support vector machines

Main article: Support vector machines

Support vector machines (SVMs) are a set of related supervised learning methods used for classification and regression. Given a set of training examples, each marked as belonging to one of two categories, an SVM training algorithm builds a model that predicts whether a new example falls into one category or the other.

Clustering

Main article: Cluster analysis

Cluster analysis or clustering is the assignment of a set of observations into subsets (called *clusters*) so that observations in the same cluster are similar in some sense. Clustering is a method of unsupervised learning, and a common technique for statistical data analysis.

Bayesian networks

Main article: Bayesian network

A Bayesian network, belief network or directed acyclic graphical model is a probabilistic graphical model that represents a set of random variables and their conditional independencies via a directed acyclic graph (DAG). For example, a Bayesian network could represent the probabilistic relationships between diseases and symptoms. Given symptoms, the network can be used to compute the probabilities of the presence of various diseases. Efficient algorithms exist that perform inference and learning.

Reinforcement learning

Main article: Reinforcement learning

Reinforcement learning is concerned with how an *agent* ought to take *actions* in an *environment* so as to maximize some notion of long-term *reward*. Reinforcement learning algorithms attempt to find a *policy* that maps *states* of the world to the actions the agent ought to take in those states. Reinforcement learning differs from the supervised learning problem in that correct input/output pairs are never presented, nor sub-optimal actions explicitly corrected.

Applications

Applications for machine learning include machine perception, computer vision, natural language processing, syntactic pattern recognition, search engines, medical diagnosis, bioinformatics, brain-machine interfaces and cheminformatics, detecting credit card fraud, stock market analysis, classifying DNA sequences, speech and handwriting recognition, object recognition in computer vision, game playing, software engineering, adaptive websites, robot locomotion, and structural health monitoring.

Machine learning techniques helped win a major software competition: In 2006, the online movie company Netflix held the first "Netflix Prize" competition to find a program to better predict user preferences and beat its existing Netflix movie recommendation system by at least 10%. The AT&T Research Team BellKor won over several other teams with their machine learning program called Pragmatic Chaos. After winning several minor prizes, it won the 2009 grand prize competition for $1 million.[3]

Software

RapidMiner, KNIME, Weka, ODM, Shogun toolbox and Orange are software suites containing a variety of machine learning algorithms.

Journals and conferences

- *Machine Learning* (journal)
- *Journal of Machine Learning Research*
- *Neural Computation* (journal)
- International Conference on Machine Learning (ICML) (conference)
- Neural Information Processing Systems (NIPS) (conference)
- List of upcoming conferences in Machine Learning and Artificial Intelligence (http://sites.google.com/site/fawadsyed/upcoming-conferences) (conference)

See also

- Computational intelligence
- Data mining
- Explanation-based learning
- Important publications in machine learning
- Multi-label classification
- Pattern recognition
- Predictive analytics

References

1. ^ Tom M. Mitchell (1997) Machine Learning p.2
2. ^ Christopher M. Bishop (2006) *Pattern Recognition and Machine Learning*, Springer ISBN 0-387-31073-8.
3. ^ "BelKor Home Page" (http://www2.research.att.com/~volinsky/netflix/) research.att.com

Further reading

- Sergios Theodoridis, Konstantinos Koutroumbas (2009) "Pattern Recognition", 4th Edition, Academic Press, ISBN 978-1-59749-272-0.
- Ethem Alpaydın (2004) *Introduction to Machine Learning (Adaptive Computation and Machine Learning)*, MIT Press, ISBN 0262012111
- Bing Liu (2007), *Web Data Mining: Exploring Hyperlinks, Contents and Usage Data (http://www.cs.uic.edu/~liub/WebMiningBook.html)* . Springer, ISBN 3540378812
- Toby Segaran, *Programming Collective Intelligence*, O'Reilly ISBN 0-596-52932-5
- Ray Solomonoff, "An Inductive Inference Machine (http://world.std.com/~rjs/indinf56.pdf) " A privately circulated report from the 1956 Dartmouth Summer Research Conference on AI.
- Ray Solomonoff, *An Inductive Inference Machine*, IRE Convention Record, Section on Information Theory, Part 2, pp., 56-62, 1957.
- Ryszard S. Michalski, Jaime G. Carbonell, Tom M. Mitchell (1983), *Machine Learning: An Artificial Intelligence Approach*, Tioga Publishing Company, ISBN 0-935382-05-4.
- Ryszard S. Michalski, Jaime G. Carbonell, Tom M. Mitchell (1986), *Machine Learning: An Artificial Intelligence Approach, Volume II*, Morgan Kaufmann, ISBN 0-934613-00-1.
- Yves Kodratoff, Ryszard S. Michalski (1990), *Machine Learning: An Artificial Intelligence Approach, Volume III*, Morgan Kaufmann, ISBN 1-55860-119-8.
- Ryszard S. Michalski, George Tecuci (1994), *Machine Learning: A Multistrategy Approach*, Volume IV, Morgan Kaufmann, ISBN 1-55860-251-8.
- Bhagat, P.M. (2005). *Pattern Recognition in Industry*, Elsevier. ISBN 0-08-044538-1.
- Bishop, C.M. (1995). *Neural Networks for Pattern Recognition*, Oxford University Press. ISBN 0-19-853864 -2.
- Richard O. Duda, Peter E. Hart, David G. Stork (2001) *Pattern classification* (2nd edition), Wiley, New York, ISBN 0-471-05669-3.
- Huang T.-M., Kecman V., Kopriva I. (2006), Kernel Based Algorithms for Mining Huge Data Sets, Supervised, Semi-supervised, and Unsupervised Learning (http://learning-from-data.com) , Springer-Verlag, Berlin, Heidelberg, 260 pp. 96 illus., Hardcover, ISBN 3-540-31681-7.
- KECMAN Vojislav (2001), Learning and Soft Computing, Support Vector Machines, Neural Networks and Fuzzy Logic Models (http://support-vector.ws) , The MIT Press, Cambridge, MA, 608 pp., 268 illus., ISBN 0 -262-11255-8.
- MacKay, D.J.C. (2003). *Information Theory, Inference, and Learning Algorithms (http://www.inference.phy.cam.ac.uk/mackay/itila/)* , Cambridge University Press. ISBN 0-521-64298-1.
- Mitchell, T. (1997). *Machine Learning*, McGraw Hill. ISBN 0-07-042807-7.
- Ian H. Witten and Eibe Frank *Data Mining: Practical machine learning tools and techniques* Morgan Kaufmann ISBN 0-12-088407-0.
- Sholom Weiss and Casimir Kulikowski (1991). *Computer Systems That Learn*, Morgan Kaufmann. ISBN 1-55860-065-5.
- Mierswa, Ingo and Wurst, Michael and Klinkenberg, Ralf and Scholz, Martin and Euler, Timm: *YALE: Rapid Prototyping for Complex Data Mining Tasks*, in Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (KDD-06), 2006.
- Trevor Hastie, Robert Tibshirani and Jerome Friedman (2001). *The Elements of Statistical Learning (http://www-stat.stanford.edu/~tibs/ElemStatLearn/)* , Springer. ISBN 0387952845.
- Vladimir Vapnik (1998). *Statistical Learning Theory*. Wiley-Interscience, ISBN 0471030031.

External links

- Ruby implementations of several machine learning algorithms (http://ai4r.rubyforge.org)
- Andrew Ng's Stanford lectures and course materials (http://see.stanford.edu/see/courseinfo.aspx?coll=348ca38a-3a6d-4052-937d-cb017338d7b1)
- The Encyclopedia of Computational Intelligence (http://scholarpedia.org/article/Encyclopedia_of_Computational_Intelligence)
- International Machine Learning Society (http://machinelearning.org/)

- Kmining List of machine learning, data mining and KDD scientific conferences (http://kmining.com/info_conferences.html)
- Machine Learning Open Source Software (http://mloss.org/about/)
- Machine Learning Video Lectures (http://videolectures.net/Top/Computer_Science/Machine_Learning/)
- Open Source Artificial Learning Software (http://salproject.codeplex.com)
- The Computational Intelligence and Machine Learning Virtual Community (http://cimlcommunity.org/)
- R Machine Learning Task View (http://cran.r-project.org/web/views/MachineLearning.html)
- Machine Learning Links and Resources (http://www.reddit.com/r/MachineLearning/)

Retrieved from "http://en.wikipedia.org/wiki/Machine_learning"
Categories: Learning in computer vision | Machine learning | Learning | Cybernetics

EXHIBIT 9

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application No.: 09/597,975 | Docket No.: UTO-101

Filing Date: 06/20/2000 | Art Unit: 2157

Applicants: Konig *et al.* | Examiner: Bharat Barot

Title: Automatic, Personalized Online Information and Product Services

CERTIFICATE OF MAILING

I hereby certify that this correspondence is being deposited with the United States Postal Service with sufficient postage as First Class Mail in an envelope addressed to: Commissioner of Patents, Alexandria, VA 22313-1450

on 8/4/05 (Date) Sylvia Lee (Signature)

SYLVIA LEE

Type or print name of person signing

Reply under 37 CFR 1.111

Commissioner for Patents
Mail Stop Non-Fee Amendment
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In reply to the Non-Final Office Action mailed by the USPTO on July 8th, 2005, the Applicant respectfully submits the following remarks.

REMARKS

Phone Interview

A phone interview took place on the August 3rd 2005 between Examiner Bharat Barot and undersigned Ron Jacobs discussing *Gerace* (U.S. Patent No. 5,991,735) in light of the independent claims 1 and 32. A conclusion was reached that *Gerace* is different from the independent claims 1 and 32 and is in fact not anticipating, teaching or suggesting the combination of elements in independent claims 1 and 32. Specifically, *Gerace* does not teach or suggest the combination of elements as listed in independent claims 1 and 32:

- estimating parameters of a learning machine, wherein the parameters define a User Model specific to the user and wherein the parameters are estimated in part from the user-specific data files (*independent claim elements 1(c) and 32(c))*
- analyzing a document *d* to identify properties of the document (*independent claim elements 1(d) and 32(d))*
- estimating a probability *P(u|d)* that an unseen document *d* is of interest to the user *u*, wherein the probability *P(u|d)* is estimated by applying the identified properties of the document to the learning machine having the parameters defined by the User Model (*independent claim elements 1(e) and 32(e))* and
- using the estimated probability to provide automatic, personalized information services to the user (*independent claim elements 1(f) and 32(f)*).

Claims Rejections, 35 USC Paragraph 102(e)

Claims 1-15, 20, 22, 22-24, 27-46, 51, 53-55 and 58-62 were rejected under U.S.C. 102(e) as being anticipated by *Gerace* (U.S. Patent No. 5,991,735).

In reply, the Applicant respectfully disagrees.

Gerace teaches:

(*Abstract*) "Based on regression analysis of **recorded responses of a first set of users** viewing the advertisements, the target user profile is refined." [underline and bold by Applicant]

(*Column 2, lines 19-20*) "...a **history** and/or pattern of user activity which in turn is interpreted as a user's habit and /or preferences." [underline and bold by Applicant]

(*Column 2, line 45*) "that records **history of users** viewing the advertisements." [underline and bold by Applicant]

(*Column 2, lines 50-53*) "... performs a regression analysis on the **recorded history of users** viewing the ads. The subroutine refines profiles of target users based on the regression analysis." [underline and bold by Applicant]

(*Claim 8*) "... **records history of users** viewing the advertisements ...". [underline and bold by Applicant]

(*Claim 9*) "... regression analysis on the **history of users** viewing the advertisements ...". [underline and bold by Applicant]

As a person of average skill in the art readily appreciates; in particular reading the above referenced sections, *Gerace* uses memorization to determine a profile of a user. *Gerace* does not teach nor suggest generalization beyond the recorded history or memorized information. Furthermore, *Gerace's* user interest is defined in a fixed set of categories

(also referred to as agate information, e.g. *sports*) and does not extend beyond the fixed set of categories (e.g. *stocks* instead of *sports*). *Gerace's* teaching is concerned with finding similar user(s), among the existing set of users with a fixed set of categories. By having a set of users that clicked or viewed an Ad that was served to them *Gerace* finds similar users (i.e. user(s) that like similar categories within the fixed set of categories) to serve them that Ad. If the AD or document belongs to a category X that is not listed or not part of the set of existing users, then *Gerace's* system has to present this Ad or unseen document to a random set of users until sufficient statistics about the users that like this has emerged. In other words, it is not taught nor is it suggested how the first set of users or the first user are/is presented with an unseen document or an unseen Ad. *Gerace* has no answer to that problem!

Accordingly, it is noted that *Gerace does not* and *can not* estimate posterior probability P(u|d) that an unseen document is of interest to a user (See *independent claim elements 1(e) and 32(e))*.

Estimating the posterior probability P(u|d) that an unseen document is of interest to a user (See *independent claim elements 1(e) and 32(e))* is just one of the elements of the claimed invention of the present application. In that light, it is noted that the way the claimed invention establishes the posterior probability P(u|d) of an unseen document is not taught nor suggested by the prior art of record. More specifically, the prior art of record does not teach or suggest a learning machine assisting in estimating P(u|d) that an unseen document is of interest to user *d* (*independent claim elements 1(e) and 32(e))*.

Furthermore, the prior art of record does not teach or suggest the step of estimating parameters of that learning machine and further assisting in estimating P(u|d) that an unseen document is of interest to user *d*. (*independent claim elements 1(c) and 32(c)).*

In summary, the Applicant submits that claims 1-15, 20, 22, 22-24, 27-46, 51, 53-55 and 58-62 are not anticipated and not suggested by *Gerace*. It is kindly requested that the claimed invention is interpreted as the combination of elements listed in each independent claim, i.e., 1(a)-1(f) and 32(a)-32(f). Accordingly, allowance of claims 1-15, 20, 22, 22-24, 27-46, 51, 53-55 and 58-62 is kindly requested.

Claims Rejections, 35 USC Paragraph 103

Claims 16-18, 47-49 were rejected under U.S.C. 103 as being unpatentable over *Gerace* (U.S. Patent No. 5,991,735).

In reply, the Applicant respectfully disagrees for the above mentioned reasons and arguments. The Applicant submits that claims 16-18, 47-49 are not suggested by *Gerace*. Accordingly, allowance of claims 16-18, 47-49 is kindly requested.

Claims Objections (Allowable Subject Matter)

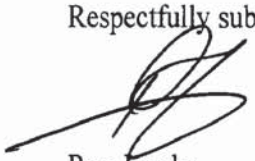
Claims 19, 21, 25-26, 50, 52 and 56-57 were objected to as being dependent upon a rejected base claim, but would be allowable if rewritten in independent form including all of the limitations of the base claim and any intervening claims.

In reply, the Applicant appreciates and thanks the Examiner for indicating allowable subject matter.

CONCLUSION

The Applicant submits that claims 1-62 are novel and unobvious over *Gerace*. In general, the Applicant submits that claims 1-62 are novel and unobvious over the prior art of record. In that light, the Applicant incorporates in this reply all previously made arguments and remarks addressing the prior art of record. Accordingly, allowance of the claims now in the application is kindly requested.

Respectfully submitted,



Ron Jacobs
Reg. No. 50,142
LUMEN Intellectual Property Services
2345 Yale Street, 2nd Floor
Palo Alto, CA 94306-1429

Phone: (650) 424-0100
Fax: (650) 424-0141
Email: ron@lumen.com

#7 OFFICIAL
RUB 12-31-03
PATENT

RECEIVED
CENTRAL FAX CENTER
DEC 16 2003

IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Application No.:	09/597,975	Atty. Docket No.:	UTO-101
Filing Date:	06/20/2000	Art Unit:	2157
Applicants:	Yochai Konig *et al.*	Examiner:	Barbara N. Burgess

Title: AUTOMATIC, PERSONALIZED ONLINE INFORMATION AND PRODUCT SERVICES

CERTIFICATE OF TRANSMISSION

I hereby certify that this correspondence is being facsimile transmitted to the U.S. Patent and Trademark Office (Fax No. 703-872-9306) on December 16, 2003.

Tianhua Gu

Typed or printed name of person signing this certificate

Tianhua Gu

Signature

REQUEST TO WITHDRAW FINALITY OF THE OFFICE ACTION

Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

Sir:

In response to the Office action mailed on December 3, 2003 and the telephonic communication with the examiner on December 12, 2003, applicants respectfully request that the finality of the Office action be withdrawn in view of the following remarks.

LUMEN INTELLECTUAL PROPERTY SERVICES
2345 Yale St., 2nd Floor
Palo Alto, CA 94306
Phone: (650) 424-0100
Fax: (650) 424-0141
gu@lumen.com
www.lumen.com

FAX COVER SHEET

Total Pages (including cover): 6

Date: December 16, 2003

From: Katharina Wang Schuster, Reg. No. 50,000

To: Attention: Examiner's Supervisor Ario Ettinene

Examiner Barbara N Burgess

Your Fax: 703-872-9306

Re: 09/597,975, (UTO-101)

Memo: Enclosed is "Request to Withdraw Finality of the Office Action" (5 pages).

This is a "FORMAL REQUEST FOR ENTRY."

Certificate of Transmission under 37 CFR 1.8
I hereby certify that this correspondence is being facsimile transmitted to the United States Patent and Trademark Office
on. 12/16/03
Date Tiantina Gu
Signature Tiantina Gu
Typed or printed name of person signing Certificate

REMARKS

Claims 1-62 are pending. Claims 1-62 were finally rejected under 35 U.S.C. § 102(e) as being anticipated by Breese *et al.* (U.S. Pat. No. 6,006,218, hereinafter referred to as "Breese").

The Finality of the Office Action Was Improper and Should be Withdrawn

MPEP § 706.07 states:

> "Before final rejection is in order a clear issue should be developed between the examiner and applicant. To bring the prosecution to as speedy conclusion as possible and at the same time to deal justly by both the applicant and the public, the invention as disclosed and claimed should be thoroughly searched in the first action and the references fully applied; and in reply to this action the applicant should amend with a view to avoiding all the grounds of rejection and objection."

The first Office action applied Breese as anticipating every one of the elements/claim limitations recited in all 62 claims. In reply to the first Office action and in accordance with the controlling case laws, *infra*, applicants pointed out, with respect to pertinent claim limitations, what Breese does **not** disclose or suggest. No claim amendments were presented in the previous Reply because original claims recite novel elements/limitations sufficient to distinguish Breese.

MPEP § 2131 states:

> "A claim is anticipated only if each and every element as set forth in the claim is found, either expressly or inherently describe, in a single prior art reference." *Verdegaal Bros. V. Union Oil Co. of California*, 814 F.2d 628, 631, 2 USP2d 1051, 1053 (Fed. Cir. 1987). "The identical invention must be shown in as complete detail as is contained in the ... claim." *Richardson v. Suzuki Motor Co.*, 868 F.2d 1226, 1236, 9 USPQ2d 1913, 1920 (Fed. Cir. 1989).

Applicants respectfully submit that Breese simply does **not** show or suggest an identical invention in as complete detail as is contained in the claims as set forth in the present application. At the minimum, Breese failed to teach or suggest claim limitations such as "**estimating parameters of a learning machine, wherein the parameters define a**

User Model ...," as explicitly recited in independent claims 1 and 32. This is particularly pointed out in the previous Reply, which is incorporated herein by reference.

> MPEP § 706.07 states:
>
> "In making the final rejection, all outstanding grounds of rejection of record should be carefully reviewed, and any such grounds relied on in the final rejection should be carefully reviewed, and any such grounds relied on in the final rejection should be reiterated. They must also be clearly developed to such an extent that applicant may readily judge the advisability of an appeal unless a single previous Office action contains a complete statement supporting the rejection.

Applicants respectfully submit that the finality was premature inasmuch as there **remain outstanding grounds of rejection of record not clearly developed to such an extent that applicants may readily judge the advisability of an appeal.** For example, independent claim 1 recites "estimating parameters of a learning machine, wherein the parameters define a User Model..." There are three limitations here, "a learning machine," "parameters," and "a User Model." All three limitations, as well as the deterministic relationship among them (i.e., the User Model is defined by the parameters of the learning model) *must* be present in Breese for an anticipatory type of rejection to stand. The cited columns of Breese refer to a database (storage) that has information (stored data) about the user and the user's interests [Office action, page 14, 2nd para.]. It is not clear at all how such a database *anticipates* or is *identical* to the claimed "User Model," which, according to the particular teaching of the present application, is a function defined by a set of parameters of a learning machine [Spec. page 14, 2nd para.; Fig. 3].

A rejection under 35 U.S.C. § 102(e) simply does not stand if the reference relied upon fails to disclose, either expressly or inherently, an identical invention in as complete detail as contained in the claims, *supra*. Thus, to obviate the 102(e) rejections, **applicants particularly pointed out, on pages 2-9 of the previous Reply, the specific limitations of the claims not disclosed in Breese,** e.g., "analyzing a document *d* to identify properties of the document," "selecting in a group of users an expert user in an area of expertise," "finding an expert User Model among User Models of the group of users," "initializing the User Model by selecting a set of predetermined

parameters of a prototype user selected by the user."

Clearly, these specific arguments do not amount to a general allegation, as the Office action has alleged. Contrary, they clearly show that, by pointing out what Breese does not teach or suggest, the language of the claims patentably distinguish them from Breese, in compliance with 37 CFR 1.111(b). Therefore, at least the aforementioned claim limitations should have been considered.

Since the final Office action did not take into consideration of these claim limitations which have been submitted to be not disclosed and not anticipated by Breese, the finality of the Office action is submitted to be premature and should be withdrawn.

> "The applicant who is seeking to define his or her invention in claims that will give him or her the patent protection to which he or she is justly entitled should receive the cooperation of the examiner to that end, and not be prematurely cut off in the prosecution of his or her application," *id.*

Since the final rejection did not include a rebuttal of all arguments raised in Applicants' previous Reply with respect to the claim limitations not disclosed in Breese, Applicants are unable to develop a clear issue or readily judge the advisability of an appeal.

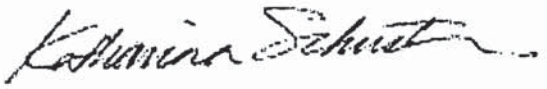
> "The examiner should never lose sight of the fact that in every case the applicant is entitled to a full and fair hearing, and that a clear issue between applicant and examiner should be developed, if possible, before appeal." MPEP 706.07.
>
> "The examiner must ... address any arguments presented by the applicant which are still relevant to any references being applied." MPEP 707.07.

In view of the foregoing, applicants therefore respectfully request that the examiner withdraws the finality of the Office action.

Applicants further respectfully submit that claims 1-62 as originally filed recite subject matter not reached by Breese under 35 U.S.C. 102(e) and are therefore allowable. The present Request is a bona fide attempt to forward the present application to allowance.

The examiner is earnestly invited to telephone the undersigned at 650-331-8413 to discuss matters pertaining to the present application or an examiner's Amendment. **Any suggested actions that would accelerate prosecution and move the present application to a condition for allowance are much appreciated.**

Respectfully submitted,



Katharina Wang Schuster, Reg. No. 50,000
Attorney for the Applicants under 37 CFR 1.34

LUMEN INTELLECTUAL PROPERTY SERVICES
2345 Yale Street, Second Floor
Palo Alto, CA 94306
(O) 650-424-0100 x 8413 (F) 650-424-0141

EXHIBIT 10

ALAN FREEDMAN

The Computer Desktop Encyclopedia

Second Edition

AMACOM
American Management Association
New York • Atlanta • Boston • Chicago • Kansas City • San Francisco • Washington, D.C.
Brussels • Toronto • Mexico City

Library of Congress Cataloging-in-Publication Data

Freedman, Alan
Computer desktop encyclopedia / Alan Freedman. -- 2nd ed.
p. cm.
ISBN 0-8144-7985-5
1. Computers--Dictionaries. I. Title.
QA76.15.F732 1999
004'.03--dc21 98-32408
CIP

Printing number

10 9 8 7 6 5 4 3 2 1

data entry department
The part of the datacenter where the data entry terminals and operators are located.

data entry operator
A person who enters data into the computer via keyboard or other reading or scanning device.

data entry program
An application program that accepts data from the keyboard or other input device and stores it in the computer. It may be part of an application that also provides updating, querying and reporting.

The data entry program establishes the data in the database and should test for all possible input errors. See *validity checking, table lookup, check digit* and *intelligent database.*

data error
Data on a digital medium has been corrupted. The error can be as little as one bit.

data file
A collection of data records. This term may refer specifically to a database file that contains records and fields in contrast to other files such as a word processing document or spreadsheet. Or, it may refer to a file that contains any type of information structure including documents and spreadsheets in contrast to a program file.

data flow
(1) In computers, the path of data from source document to data entry to processing to final reports. Data changes format and sequence (within a file) as it moves from program to program.

(2) In communications, the path taken by a message from origination to destination and includes all nodes through which the data travels.

data flow diagram
A description of data and the manual and machine processing performed on the data.

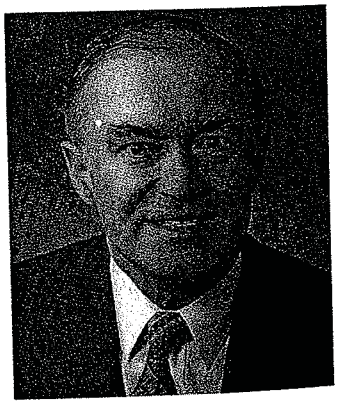
data fork
The part of a Macintosh file that contains data. For example, in a HyperCard stack, text, graphics and HyperTalk scripts reside in the data fork, while fonts, sounds, control information and external functions reside in the resource fork.

data format
Same as *file format.*

Data General
(Data General Corporation, Westboro, MA, www.dg.com) A computer manufacturer founded in 1968 by Edson de Castro. In 1969, it introduced the Nova, the first 16-bit mini with four accumulators, a leading technology at the time. During its early years, the company was successful in the scientific, academic and OEM markets. With its 32-bit ECLIPSE family of computers and its Comprehensive Electronic Office (CEO) software, Data General gained entry into the commercial marketplace in the early 1980s.

In 1989, the company introduced its AViiON line of UNIX-based servers that use the Motorola 88000 CPU, and more powerful models continue to be introduced. Data General's CLARiiON line of fault-tolerant (RAID) storage systems, introduced in 1992, are available for UNIX-based IBM and Sun computer systems.

The "Eagle project," DG's development of its ECLIPSE and first 32-bit computer, was chronicled in Tracy Kidder's Pulitzer-prize winning novel, "Soul of a New Machine," published by Little, Brown and Company, ISBN 0-316-49170-5.



Edson de Castro
De Castro founded Data General as the minicomputer market began to flourish. His line of Nova machines helped expand the market for low-priced (under $100,000) computers. This was a time when minicomputers were expected to make mainframes obsolete. *(Courtesy of Data General Corporation.)*

EXHIBIT 11

FULLY REDACTED

EXHIBIT 12

Konig

Bayesian Statistics:

an introduction

PETER M. LEE
Provost of Wentworth College, University of York, England

A CHARLES GRIFFIN BOOK

OXFORD UNIVERSITY PRESS
New York

Edward Arnold
A division of Hodder & Stoughton
LONDON MELBOURNE AUCKLAND

To My Mother
and
to the Memory of My Father

First published in Great Britain 1989

British Library Cataloguing in Publication Data

Lee, P.M. (Peter M)
Bayesian statistics
1. Statistical analysis. Bayesian theories
I. Title
519.5'42

ISBN 0–85264–309–8 (paper)
ISBN 0–85264–298–9 (boards)

Typeset in 10½/12 pt Times by Wearside Tradespools, Fulwell, Sunderland
Printed and bound in Great Britain for Edward Arnold, the educational, academic and medical publishing division of Hodder and Stoughton Limited, 41 Bedford Square, London WC1B 3DQ by Richard Clay Plc, Bungay, Suffolk

---

Published in the USA by
Oxford University Press
200 Madison Avenue, New York, NY 10016

Library of Congress Cataloguing-in-Publication Data

Available on request from Oxford University Press

OUP ISBN 0–19–520802–1 (Cloth)
ISBN 0–19–520803–X (Paper)

Printed in Great Britain

2

Bayesian Inference for the Normal Distribution

2.1 Nature of Bayesian inference

Preliminary remarks

In this section a general framework for Bayesian statistical inference will be provided. In broad outline we take prior beliefs about various possible hypotheses and then modify these prior beliefs in the light of relevant data which we have collected in order to arrive at posterior beliefs. (The reader may prefer to return to this section after reading the next section, which deals with one of the simplest special cases of Bayesian inference.)

Post is prior times likelihood

Almost all of the situations we will think of in this book fit into the following pattern. Suppose that you are interested in the values of k unknown quantities

$$\boldsymbol{\theta} = (\theta_1, \theta_2, \ldots, \theta_k)$$

(where k can be one or more than one) and that you have some *a priori* beliefs about their values which you can express in terms of the p.d.f.

$$p(\boldsymbol{\theta}).$$

Now suppose that you then obtain some data relevant to their values. More precisely, suppose that we have n observations

$$\boldsymbol{X} = (X_1, X_2, \ldots, X_n)$$

which have a probability distribution that depends on these k unknown quantities as parameters, so that the p.d.f. (continuous or discrete) of the vector $\boldsymbol{X}$ depends on the vector $\boldsymbol{\theta}$ in a known way. Usually the components of $\boldsymbol{\theta}$ and $\boldsymbol{X}$ will be integers or real numbers, so that the components of $\boldsymbol{X}$ are random variables, and so the dependence of $\boldsymbol{X}$ on $\boldsymbol{\theta}$ can be expressed in terms of a p.d.f.

$$p(\boldsymbol{X}|\boldsymbol{\theta}).$$

You then want to find a way of expressing your beliefs about $\boldsymbol{\theta}$ taking into account both your prior beliefs and the data. Of course, it is possible that your prior beliefs about $\boldsymbol{\theta}$ may differ from mine, but very often we will agree on the way in which the data are related to $\boldsymbol{\theta}$ (that is, on the form of $p(\boldsymbol{X}|\boldsymbol{\theta})$). If this is so, we will differ in our posterior beliefs (i.e. in our beliefs after we have obtained the data), but it will turn out that if we can collect enough data, then our posterior beliefs will usually become very close.

The basic tool we need is Bayes' Theorem for random variables (generalized to deal with random vectors). From this theorem we know that

$$p(\boldsymbol{\theta}|\boldsymbol{X}) \propto p(\boldsymbol{\theta})p(\boldsymbol{X}|\boldsymbol{\theta}).$$

Now we know that $p(\boldsymbol{X}|\boldsymbol{\theta})$ considered as a function of $\boldsymbol{X}$ for fixed $\boldsymbol{\theta}$ is a density, but we will find that we often want to think of it as a function of $\boldsymbol{\theta}$ for fixed $\boldsymbol{X}$. When we think of it in that way it does not have quite the same properties—for example, there is no reason why it should sum (or integrate) to unity. Thus in the extreme case where $p(\boldsymbol{X}|\boldsymbol{\theta})$ turns out not to depend on $\boldsymbol{\theta}$, then it is easily seen that it can quite well sum (or integrate) to ∞. When we are thinking of $p(\boldsymbol{X}|\boldsymbol{\theta})$ as a function of $\boldsymbol{\theta}$ we call it the *likelihood* function. We sometimes write

$$l(\boldsymbol{\theta}|\boldsymbol{X}) = p(\boldsymbol{X}|\boldsymbol{\theta}).$$

Just as we sometimes write $p_{\boldsymbol{X}|\boldsymbol{\theta}}(\boldsymbol{X}|\boldsymbol{\theta})$ to avoid ambiguity, if we really need to avoid ambiguity we write

$$l_{\boldsymbol{\theta}|\boldsymbol{X}}(\boldsymbol{\theta}|\boldsymbol{X})$$

but this will not usually be necessary. Sometimes it is more natural to consider the *log-likelihood* function

$$L(\boldsymbol{\theta}|\boldsymbol{X}) = \log l(\boldsymbol{\theta}|\boldsymbol{X}).$$

With this definition and the definition of $p(\boldsymbol{\theta})$ as the prior p.d.f. for $\boldsymbol{\theta}$ and of $p(\boldsymbol{\theta}|\boldsymbol{X})$ as the posterior p.d.f. for $\boldsymbol{\theta}$ given $\boldsymbol{X}$, we may think of Bayes' Theorem in the more memorable form

$$\text{Posterior} \propto \text{Prior} \times \text{Likelihood}.$$

This relationship summarizes the way in which we should modify our beliefs in order to take into account the data we have available.

Likelihood can be multiplied by any constant

We note that, because of the way we write Bayes' Theorem with a proportionality sign, it does not alter the result if we multiply $l(\boldsymbol{\theta}|\boldsymbol{X})$ by

any constant or indeed more generally by anything which is a function of $\boldsymbol{X}$ alone. Accordingly, we can regard the definition of the likelihood as being *any constant multiple* of $p(\boldsymbol{X}|\boldsymbol{\theta})$ rather than necessarily equalling $p(\boldsymbol{X}|\boldsymbol{\theta})$ (and similarly the log-likelihood is undetermined up to an additive constant). Sometimes the integral

$$\int l(\boldsymbol{\theta}|\boldsymbol{X})\,\mathrm{d}\boldsymbol{\theta}$$

(interpreted as a multiple integral $\int\int \ldots \int \ldots \mathrm{d}\theta_1\mathrm{d}\theta_2 \ldots \mathrm{d}\theta_k$ if $k>1$ and interpreted as a summation or multiple summation in the discrete case), taken over the admissible range of $\boldsymbol{\theta}$, is finite, although we have already noted that this is not always the case. When it is, it is occasionally convenient to refer to the quantity

$$\frac{l(\boldsymbol{\theta}|\boldsymbol{X})}{\int l(\boldsymbol{\theta}|\boldsymbol{X})\,\mathrm{d}\boldsymbol{\theta}}.$$

We shall call this the *standardized likelihood*, that is, the likelihood scaled so that the area, volume or hypervolume under the curve, surface or hypersurface is unity.

Sequential use of Bayes' Theorem

We should also note that the method can be applied *sequentially*. Thus, if you have an initial sample of observations $\boldsymbol{X}$, you have

$$p(\boldsymbol{\theta}|\boldsymbol{X}) \propto p(\boldsymbol{\theta})\,l(\boldsymbol{\theta}|\boldsymbol{X}).$$

Now suppose that you have a second set of observations $\boldsymbol{Y}$ distributed independently of the first sample. Then

$$p(\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{Y}) \propto p(\boldsymbol{\theta})\,l(\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{Y}).$$

But independence implies

$$p(\boldsymbol{X}, \boldsymbol{Y}|\boldsymbol{\theta}) = p(\boldsymbol{X}|\boldsymbol{\theta})p(\boldsymbol{Y}|\boldsymbol{\theta})$$

from which it is obvious that

$$l(\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{Y}) \propto l(\boldsymbol{\theta}|\boldsymbol{X})\,l(\boldsymbol{\theta}|\boldsymbol{Y})$$

and hence

$$\begin{aligned} p(\boldsymbol{\theta}|\boldsymbol{X}, \boldsymbol{Y}) &\propto p(\boldsymbol{\theta})\,l(\boldsymbol{\theta}|\boldsymbol{X})\;\, l(\boldsymbol{\theta}|\boldsymbol{Y}) \\ &\propto p(\boldsymbol{\theta}|\boldsymbol{X})\,l(\boldsymbol{\theta}|\boldsymbol{Y}). \end{aligned}$$

So we can find your posterior for $\boldsymbol{\theta}$ given $\boldsymbol{X}$ and $\boldsymbol{Y}$ by treating your posterior given $\boldsymbol{X}$ as the prior for the observation $\boldsymbol{Y}$. This formula will work *irrespective of the temporal order* in which $\boldsymbol{X}$ and $\boldsymbol{Y}$ are observed.