Diplomarbeit

Zeitreihenprognose für Warenwirtschaftssysteme unter Berücksichtigung asymmetrischer Kostenfunktionen

Stefan Rüping

Diplomarbeit
am Fachbereich Informatik
der Universität Dortmund

22. September 1999

Betreuer:

Prof. Dr. Katharina Morik
Dipl.-Inform. Thorsten Joachims

Inhaltsverzeichnis

Abbildungsverzeichnis

Kapitel 1

Einleitung

Viele moderne Unternehmen besitzen ein weitverteiltes Netz von Filialen, Produktionsstätten und Lägern und sind auf enge Weise mit ihren Kunden und Lieferanten verknüpft. Dabei werden Waren oft über weite Strecken zwischen Verkäufer und Käufer oder zwischen Zentrallager und Filiale transportiert und zwischengelagert. Daher ist die Logistik ein wichtiger Kostenfaktor in modernen Unternehmen. Sie ist sowohl dafür verantwortlich, dass genügend Waren zum richtigen Zeitpunkt vorhanden sind, als auch dass keine unnötigen Kosten durch übervolle Läger entstehen.

Ziel der Unternehmen ist es, eine optimale Lagerhaltungspolitik zu bestimmen und in einem Warenwirtschaftssystem umzusetzen und zu kontrollieren. Die ständige Kontrolle der Lagervorgänge ist nötig, da wesentliche Größen des Lager- und Produktionsprozesses zufälligen Einflüssen unterworfen ist: Schwankungen in der Nachfrage oder der Lieferbarkeit der Produkte führen dazu, dass in den meisten Fällen keine in allen Fällen optimale Lagerhaltungspolitik existiert, sondern dass man sich mit einer im Erwartungswert optimalen (oder hinreichend guten) Politik zufrieden geben muss.

Es ist klar, dass die Aufstellung einer solchen Lagerhaltungspolitik als wichtigen Bestandteil die treffende Prognose der wesentlichen Größen umfasst. Man beachte, dass hier die Sprache von einer *treffenden*, nicht von einer *genauen* Prognose ist. Der Grund dafür liegt darin, dass die Prognose nicht um ihrer selbst willen erstellt wird, sondern sie erstellt wird, um in der Lagerverwaltung benutzt zu werden. Als Gütekriterium einer Prognose muss also nicht ein abstraktes mathematische Kriterium hinzugezogen werden, sondern die Verwendbarkeit der Prognose für eine optimale Bestellpolitik. Anders ausgedrückt heißt das, dass die Prognose betriebswirtschaftliche Anforderungen zu berücksichtigen hat.

Als wichtigste Anforderung an eine Prognose gilt, dass ein Über- und ein Unterschätzen des tatsächlich realisierten Wertes nicht dieselben Auswirkungen haben. Überschätzt man beispielsweise die Nachfrage einer Produktionsstätte, so wird man mehr Waren einlagern als tatsächlich gebraucht werden, wodurch einige Lagerkosten entstehen. Unterschätzt man aber die Nachfrage, so kommt es aufgrund fehlender Produktionsgüter zu einem sehr teuren Produktionsausfall. Bewertet man die Qualität einer Prognose mit einer Verlustfunktion, die aus der Differenz von vorhergesagtem und tatsächlich realisiertem Wert die entstehenden Kosten berechnet, so muss man also asymmetrische Verlustfunktionen einsetzen.

Weiterhin muss das Prognoseverfahren in der Lage sein, eine Vielzahl von Faktoren in seine Prognose mit einfließen zu lassen. Die Schwankungen von Nachfrage, Lieferbarkeit und andere Größen resultieren aus einer Vielzahl von Einflüssen, deren genaue Auswirkungen oft unbekannt sind, deren Berücksichtigung die Prognose aber doch deutlich verbessern kann. Optimal wäre also ein Prognoseverfahren, dass eine großen Menge von Attributen verarbeiten und in die Prognose einfließen lassen kann.

Ein solches Verfahren ist die Support Vector Machine (SVM). Die Support Vector Machine basiert auf der Arbeit von Vladimir Vapnik (zusammengefasst in [Vapnik, 1998]) und ist ein Lernverfahren, das effektiv zum Lernen mit großen Datenmengen und vielen Attributen genutzt werden kann. Support Vector Machines wurden z.B. erfolgreich zur Klassifikation von Texten und zur Bilderkennung eingesetzt.

Ein Vorteil der Support Vector Machine ist, dass sie erlaubt die Kapazität (Ausdrucksstärke) des Lernalgorithmus zu kontrollieren. Zwar erlaubt ein ausdrucksstarker Lernalgorithmus die gegebenen Daten sehr genau zu lernen, es besteht aber die Gefahr, dass er sich den Trainingsdaten zu stark anpasst und dadurch auf neuen Daten ein schlechteres Verhalten zeigt als ein ausdrucksschwächerer Algorithmus (Overfitting). Es ist daher sinnvoll, unter mehreren Lernalgorithmen, die die gleiche Genauigkeit liefern, den mit der geringeren Kapazität wählen. Auch in der Philosophie ist diese Prinzip bekannt. So erklärt Karl Popper in [Popper, 1989] den Vorzug einfacherer Sätze aufgrund des Kriteriums der Falsifizierbarkeit::

> Einfachere Sätze sind deshalb höher zu bewerten als weniger einfache, weil sie *mehr sagen*, weil ihr empirischer Gehalt größer ist, weil sie besser überprüfbar sind.

Die Support Vector Machine ist auch aus einem anderen Grund für die Prognose in Warenwirtschaftssystemen geeignet: Die Support Vector Machine minimiert den Fehler anhand einer explizit vorgegebenen Verlustfunktion. Bisher werden in der Regel symmetrische lineare oder quadratische Verlustfunktionen verwendet, für die theoretische Lösbarkeit des Problems muss aber lediglich die Konvexität der Verlustfunktion vorausgesetzt werden. Es sollte also möglich sein, eine Verlustfunktion zu wählen, die die in der Realität auftretenden Kosten möglichst genau abbildet.

In dieser Arbeit soll insbesondere der Fall der Prognose der Nachfrage im Einzelhandel untersucht werden. Dieses Anwendungsgebiet hat einige Besonderheiten, die spezieller behandelt werden müssen: Zum einen resultiert in diesem Fall die Nachfrage aus den einzelnen Kaufentscheidungen vieler Kunden, was bedeutet, dass mit starken Schwankungen in den Daten zu rechnen ist. Zum anderen gibt es einige Faktoren, die einen sehr starken Einfluss auf die allgemeinen Nachfrage haben und deren effektive Behandlung eine wichtige Voraussetzung für eine gute Prognose ist; zu diesen Faktoren zählen unter anderem das Auftreten von Feiertagen und Ferien.

Eine weitere Herausforderung ist die sehr dynamische Wettbewerbsituation im Einzelhandel. Aufgrund der häufigen Einführung neuer Produkte und dem sich ständig änderndem Käuferverhalten ist es nötig, schnell auf neue Trends zu reagieren und Vorhersagen aufgrund weniger oder ungenauer Daten zu machen. Eine Möglichkeit ist hier die Benutzung ähnlicher, bekannter Daten für neue Produkte. Dies führt zu dem Schlussverfahren der Transduktion, d.h. der Vorhersage neuer Daten aus bekannten Daten ohne dem Umweg über eine allgemeine, alle vorstellbaren Daten erklärende Hypothese (wie es bei der Induktion der Fall wäre).

Durch die direkte Berücksichtigung der betriebswirtschaftlichen Besonderheiten der Prognose von Zeitreihen in Warenwirtschaftssystemen ist zu erwarten, dass damit genauere Prognosen möglich sind als mit symmetrischen Verfahren, die die geforderte Asymmetrie erst in einem späteren Schritt berücksichtigen.

1.1 Fragestellung

Zusammengefasst kann man die wesentlichen Fragen dieser Diplomarbeit wie folgt darstellen:

- Lassen sich mit der Support Vector Machine auch asymmetrische Verlustfunktionen behandeln?
- Welche Besonderheiten hat die asymmetrische Prognose gegenüber einer symmetrischen?
- Wie lässt sich die Support Vector Machine auf das spezielle Anwendungsgebiet der Prognose von Zeitreihen in Warenwirtschaftssystemen anwenden?
- Wie lassen sich die speziellen Probleme der dynamischen Situation im Einzelhandel mit der SVM lösen?
- Hat der Einsatz der SVM Vorteile gegenüber anderen Prognoseverfahren?

Kapitel 2

Statistische Lerntheorie und Support Vector Machines

Beim Versuch einen Lernalgorithmus zu konstruieren spielt die Frage, was *Lernen* eigentlich ist, eine wichtige Rolle. Diese Frage ist oft diskutiert worden, und die Vorstellungen, die man vom Prozess des Lernens hat, spiegeln sich direkt in den Anforderungen einen Lernalgorithmus wieder.

Einer Definition von Simon in [Simon, 1983] zufolge kann man das Lernen wie folgt definieren:

> Learning denotes changes in the system that are adaptive in the sense that they enable the system to do the same task or tasks drawn from the same population more efficiently and more effectively the next time.

Zentral an dieser Definition ist, dass sie verlangt die Leistung des Systems zu messen um allein darüber Aussagen über einen Lernerfolg zu machen. Das Leistungsprinzip beim Lernen ist an verschiedenen Stellen kritisiert worden. Zum einen kann man die Leistung eines Systems auf verschiedene Arten messen, d.h. eine vernünftige Leistungsmessung setzt voraus, dass man das Ziel der Handlung kennt. Daher kann man auch sehr abstruse Handlungsweisen durch Definition eines passenden Ziels als Lernerfolg werten.

Der zweite Kritikpunkt gegen die Leistungsmessung beim Lernen ist der, dass ein System auch lernen kann ohne das gelernte Wissen direkt anzuwenden. Ein Roboter, der bei einer Fahrt durch ein Labyrinth eine Karte erstellt, kann zwar durch dieses Wissen die Aufgabe einen bestimmten Punkt des Labyrinths aufzusuchen besser lösen, er hat aber auch ohne jemals zu diesem Punkt zurückkehren zu müssen bereits etwas über den Aufbau des Labyrinths gelernt.

Aufgrund dessen stellt Michalski in [Michalski, 1986] die Definition

> Learning ist constructing or modifying representations of what is being experienced.

auf. Trotzdem bleibt die Leistung ein wichtiger Aspekt des Lernens. Michalski betont drei Dimensionen für die Bewertung der konstruierten Repräsentationen: Gültigkeit, Effektivität und Abstraktionsgrad. Die Gültigkeit beschreibt die Genauigkeit mit der die konstruierten Repräsentationen die Realität beschreiben. Die Effektivität beschreibt genau den Leistungsaspekt des Lernens, also die Nützlichkeit der Repräsentationen in Bezug auf ein bestimmtes Ziel. Der Abstraktionsgrad beschreibt die Detailliertheit der Konzepte in der Beschreibung und bestimmt damit, was die Repräsentation erklären kann.

Trotz der Einwände gegen die alleinige Definition des Lernerfolgs über die Leistung ist in vielen praktischen Anwendungen dieser Aspekt der wichtigste. Dies ist der Fall, wenn der Lernalgorithmus als Komponente eines größeren Systems benutzt wird um dessen Verhalten zu verbessern. Dann ist oft das Ziel des Lernens klar definiert und die Leistung objektive messbar, beispielsweise in erwirtschaftetem Gewinn oder Einsparung einer Ressource.

Zwar könnte man argumentieren, dass die Gültigkeit die Effektivität direkt beeinflusst, d.h. ein genaueres Verständnis der Realität ermöglicht eine effektivere Handlungsweise, allerdings folgt daraus nicht zwingend, dass bei der Konstruktion eines Lernalgorithmus beide Ziele im gleichen Maße berücksichtigt werden müssen. Es könnte sich nämlich herausstellen, dass dieses Lernproblem deutlich schwieriger zu lösen ist, als ein Lernproblem, dass sich hauptsächlich auf eins der Ziele konzentriert. Dies würde dazu führen, dass erheblich mehr Beispiele benötigt werden bzw. die erzielten Ergebnisse bei gleicher Anzahl von Beispielen deutlich schlechter werden.

Vielmehr ist es denkbar, zwei Lernalgorithmen auf das gleiche Problem anzusetzen, von denen der eine auf einen möglichst hohen Realitätsgrad abzielt und der andere auf eine große Effektivität. Dann könnte man in einem zweiten Schritt die Ergebnisse des ersten Algorithmus im zweiten Algorithmus zur Verbesserung der Ergebnisse berücksichtigen. Diese Trennung der Ziele bedeutet eine einfachere Implementierung der Lernalgorithmen.

Im weiteren wird hauptsächlich das Ziel der Effektivität der Lernalgorithmus weiterverfolgt und formalisiert.

Eine der Formalisierungen des Lernens ist, das Lernen als das Finden einer Funktion zu betrachten. Die Aufgabenstellung besteht dann darin, zu einer gegebenen Menge von Beispielen $(x_i, y_i) \in \mathcal{X} \times \mathcal{Y}$ eine Funktion $f : \mathcal{X} \rightarrow \mathcal{Y}$ zu finden, die ein vorgegebenes Qualitätskriterium erfüllt.

Diese Formalisierung ist insbesondere dann günstig, wenn die Anwendung des Lernergebnisses darin besteht, aufgrund eines gegebenen x eine möglichst gute Aktion auszuwählen oder Aussagen über ein zukünftiges Ereignis zu machen. Dabei ist es nämlich unerheblich, das tatsächlich hinter den Daten steckende Modell genau zu erkennen, solange nur die gewählte Aktion möglichst effektiv ist bzw. das zukünftige Ereignis möglichst gut vorhergesagt wird. Insbesondere wird nicht vorausgesetzt, dass der Zusammenhang zwischen den $\mathcal{X}$- und $\mathcal{Y}$ -Werten tatsächlich funktional ist, die Wahl einer Funktion als Modell wird allein dadurch gerechtfertigt, dass die Aufgabenstellung lediglich verlangt zu einem x-Wert *einen* y-Wert zu finden, der das Qualitätskriterium erfüllt.

Dieses Problem kann nicht ohne weitere Einschränkungen an die Menge der zu betrachtenden Funktionen gelöst werden. Eine praktische Umsetzung setzt voraus, dass das Finden einer genügend guten Zielfunktion mit einer nicht zu großen Anzahl von Beispielen und in hinreichend kurzer Zeit machbar sein muss. Formal werden diese Anforderungen zum Beispiel im Begriff der PAC-Lernbarkeit festgehalten (siehe [Kearns und Vazirani, 1994]).

2.1 Die statistische Lerntheorie

Um das allgemeine Lernproblem genauer zu beschreiben, ist es nötig das Qualitätskriterium an die Lösung zu formalisieren und anzugeben, welche Arten von Beispiele und Funktionen man betrachten möchte. Eine statistische Sichtweise ist dabei nützlich.

2.1.1 Das statistische Lernproblem

In der statistischen Formulierung des Lernproblems werden die Beispiele durch Wahrscheinlichkeitsverteilungen charakterisiert und die Menge der zu betrachtenden Funktionen $g_\alpha(x), \alpha \in \Lambda$ explizit vorgegeben. Eine Verlustfunktion definiert die Größe des Fehlers, den eine Funktion g_α auf einem Beispiel (x, y) macht. Das Qualitätskriterium besteht nun in der Minimierung des erwarteten Fehlers über alle denkbaren Beispiele.

Die Minimierung des erwarteten Risikos implementiert die Idee, das Lernergebnis möglichst effektiv zu machen. Je bedeutender die Abweichung der Vorhersage in Bezug auf das Lernziel des Algorithmus ist, und je wahrscheinlicher ein Beispiel vorkommt, desto geringer soll der Fehler sein, den der Algorithmus auf diesem Beispiel macht.

Statistisches Lernproblem: Gegeben seien Beispiele $(x_1, y_1), \ldots, (x_n, y_n) \in X \times Y$, wobei die x_i unabhängig identisch nach $F(x)$ verteilt sind und die y_i nach $F(y|x_i)$ verteilt sind. $g_\alpha(x), \alpha \in \Lambda$ sei eine Menge von Funktionen und $L : Y \times Y \rightarrow \mathbf{R}$ eine Funktion. Finde eine Funktion $g_\alpha : X \rightarrow Y$, die das erwartete Risiko

$$R(\alpha) = \int L(y, g_\alpha(x)) dF(x, y)$$

(mit $F(x, y)$ als gemeinsame Verteilung der x und y) minimiert.

Also formale Voraussetzung muss man hier die Existenz des Erwartungswertes fordern. Da die Verlustfunktion L und die Funktionenklasse g_α bereits zu Anfang bekannt sind, kann man die Definition vereinfachen indem man $Q(z, \alpha) := L(z, g_\alpha(z))$ definiert.

Wichtig in der Formulierung ist, dass lediglich die (x_i, y_i) bekannt sind, nicht die Verteilungen $F(x)$ und $F(y|x)$ selbst. Ein Lernalgorithmus der dieses Problem löst muss also gleichzeitig Information über die Verteilung der Beispiele sammeln und daraus eine Lösung zur Minimierung des erwarteten Risikos bestimmen.

Diese Formulierung des Lernproblems erlaubt, eine große Menge von Problemen so darzustellen. Durch geeignete Wahl von $F(y|x)$ als Diracmaß ($F(y|x) = \infty$ falls $y = f(x)$ und $F(y|x) = 0$ sonst) entspricht das Lernproblem z.B. dem Schätzen der Funktion f. Ansonsten ist der funktionale Zusammenhang zwischen x und y hier durch die Regressionsfunktion

$$r(x) = \int y dF(y|x)$$

gegeben. Wählt man als Verlustfunktion $L(x, g) = (y - f(x))^2$ so entspricht das statistische Lernproblem gerade dem Schätzen der Regression.

Lösungsansätze für das statistische Lernproblem

Um das Lernproblem zu lösen könnte man versuchen, die nötige Verteilung $F(x)$ zu bestimmen um daraus die Lösungsfunktion g_α direkt auszurechnen. Leider ist das Schätzen einer Verteilung aus empirischen Daten ein sehr schwieriges Problem, so dass der direkte Lösungsweg keine guten Ergebnisse verspricht. Anwendbar ist das direkte Verfahren, wenn man näherer Informationen über die in Frage kommenden Verteilungen hat. Ein klassischer Ansatz der Statistik ist beispielsweise, eine Klasse von Verteilungen vorzugeben (z.B. die Normalverteilung) und die unbekannten Parameter zu schätzen (z.B. Erwartungswert und Varianz).

In praktischen Anwendungen stellt sich aber das Problem, dass man die Verteilung der Daten nicht kennt bzw. dass die bekannten Standardverteilungen die Verteilung der Daten zwar approximieren, aber nicht identisch sind. Man hat also bereits durch die Wahl des Modells einen Fehler im Verfahren, der sich später nicht wieder ausgleichen lässt.

Das Problem der direkten Lösung des statistischen Lernproblem ist, dass man zur Bestimmung der Verteilung mehr Information über die Daten braucht, als zur Lösung des Problems nötig ist. Zur Minimierung des erwarteten Risikos genügt es nämlich, statt der Verteilung der Daten das erwartete Risiko einer Funktion aus den Daten zu schätzen. Verfahren dieser Art fasst man unter dem Begriff *empirische Risikominimierung* zusammen.

2.1.2 Empirische Risikominimierung

Die Idee der Empirischen Risikominimierung (ERM) ist, statt des erwarteten Risikos $R(\alpha)$ für eine Menge von Beispielen $(x_1, y_1), \ldots, (x_n, y_n)$ das empirische Risiko

$$R_{emp}(\alpha) = \frac{1}{n} \sum_{i=1}^{n} Q(x_i, \alpha)$$

zu betrachten. Dieses Verfahren beruht darauf, dass nach dem Gesetz der großen Zahl für eine festes $\alpha \in \Lambda$ das empirische Risiko $R_{emp}(\alpha)$ gegen das erwartete Risiko $R(\alpha)$ in Verteilung konvergiert und man hofft, dass die Funktion, die das empirische Risiko minimiert, auch das erwartete Risiko minimiert.

Das Prinzip der empirischen Risikominimierung ist die Lösung α_n zu wählen, die das empirische Risiko auf den gegebenen Beispielen $(x_1, y_1), \ldots, (x_n, y_n)$ minimiert. Dadurch stellt sich die Frage, ob mit steigender Anzahl von Beispielen das Ergebnis tatsächlich immer genauer wird. Dies führt zur Frage der *Konsistenz*.

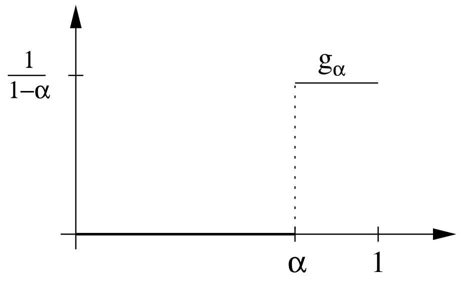


Abbildung 2.1: Die Funktion g_α.

Konsistenz der ERM: Das Prinzip der empirischen Risikominimierung heißt konsistent auf den Funktionen $Q(x,\alpha), \alpha \in \Lambda$ für die Wahrscheinlichkeitsverteilung $F(x)$, wenn gilt

$$R(\alpha_l) \stackrel{P}{\to} inf_{\alpha\in\Lambda} R(\alpha) \quad l \to \infty \tag{2.1}$$

$$R_{emp}(\alpha_l) \stackrel{P}{\to} inf_{\alpha\in\Lambda} R(\alpha) \quad l \to \infty. \tag{2.2}$$

D.h. das Risiko der Lösung der empirischen Risikominimierung muss sowohl bezüglich des beobachteten Risikos $R_{emp}(\alpha_l)$ als auch bezüglich des erwarteten Risikos $R(\alpha_l)$ gegen das minimal möglich Risiko konvergieren. Die empirische Risikominimierung muss nicht konsistent sein, wie folgendes Beispiel zeigt:

Beispiel: Für $\alpha \in [0,1[$ sei $g_\alpha(x) = 0$ falls $x \leq \alpha$ und $g_\alpha(x) = 1/(1-\alpha)$ sonst (Siehe Abbildung 2.1). $F(x)$ sei die Gleichverteilung auf $[0,1[$. Weiter sei $Q(x,\alpha) = g_\alpha(x)$. Dann ist $R_{emp}(\alpha_l) = 0$ (wähle $\alpha_l = \max\{x_1,\ldots,x_n\}$), aber

$$R(\alpha) = \int_0^1 Q(x,\alpha)dx = \int_\alpha^1 \frac{1}{1-\alpha}dx = \frac{1}{1-\alpha}(1-\alpha) = 1.$$

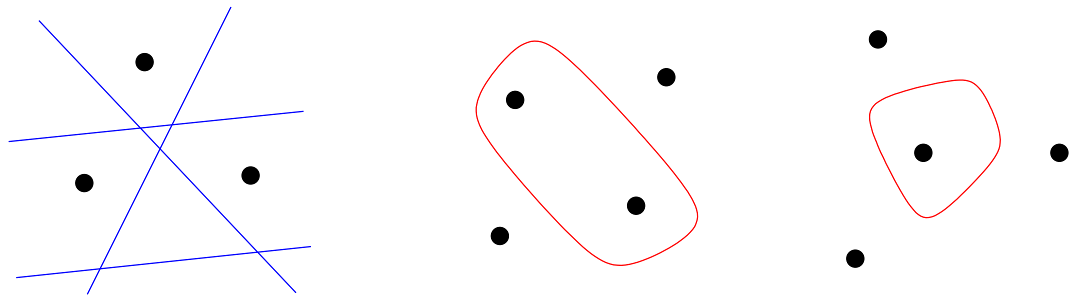
Auch wenn die empirische Risikominimierung konsistent ist, kann es in der Praxis Probleme geben. Das Problem ist, dass man in der Praxis nur kleine Beispielmengen hat, so dass das erwartete Risiko der Lösung α_l viel höher sein kann als das minimale erwartet Risiko. „klein“ ist hier relativ zu sehen zur benötigten Menge von Beispielen die man braucht, um eine kleine Abweichung zu garantieren.

Dieses Problem wir umso größer, je ausdrucksstärker (und damit komplexer) die Menge der Funktionen g_α ist. Zwar kann man mit einer ausdrucksstarken Menge von Funktionen das empirische Risiko stark minimieren, die Anzahl der Beispiele die man braucht, um einen kleine Abweichung zu garantieren wird aber deutlich größer. Man muss hier also zwischen einer einfachen und einer komplexen Lösung abwägen.

2.2 Strukturelle Risikominimierung

Die Idee der strukturellen Risikominimierung stammt von Vladimir Vapnik [Vapnik, 1995]. Die Fragestellung dabei ist, wann garantiert werden kann, dass die Lösungen α_l des empirischen Risikominimierungsprinzips für $l \to \infty$ gegen eine Lösung α^* des statistischen Lernproblems konvergieren und was über den Fehler einer Lösung α^* ausgesagt werden kann.

Dies wird durch die Konstruktion einer oberen Schranke für den erwarteten Fehler erreicht, die aus dem empirischen Fehler $R(\alpha_l)$ und einem Konfidenzintervall besteht. Für die Konstruktion des Konfidenzintervalls benötigt man den Begriff der Vapnik–Chervonenkis–Dimension.



(a) (b) (c)

Abbildung 2.2: VC-Dimension des $\mathbf{R^2}$.

2.2.1 Die Vapnik–Chervonenkis–Dimension

Die VCdim von Indikatorfunktionen

$\theta_A : X \to \{0,1\}$ heißt Indikatorfunktion, wenn es eine Teilmenge $A \subseteq X$ gibt, so dass $\theta_A(x) = 1 \Leftrightarrow x \in A$. Die Funktion θ_A teilt also die Menge X in die Klassen A und $\neg A$ auf. Insbesondere sei im folgenden $\theta : \mathbf{R} \to \{\mathbf{0},\mathbf{1}\}$ gegeben durch $\theta(x) = 1 \Leftrightarrow x \geq 0$.

Die Vapnik–Chervonenkis–Dimension einer Menge $Q(z,\alpha), \alpha \in \Lambda$ von Indikatorfunktionen ist wie folgt definiert ([Vapnik, 1998], Kapitel 4.9):

VC–Dimension: Die VC–Dimension einer Menge $Q(z,\alpha), \alpha \in \Lambda$ von Indikatorfunktionen ist die größte Anzahl von Punkten $z_1,\ldots,z_l$, die auf alle 2^l Arten in zwei Klassen aufgeteilt werden können.

Es ist lediglich gefordert, dass *eine* Menge von l Punkten sich in alle Teilklassen zerlegen lässt, für eine beliebige Menge von l Punkten kann im Allgemeinen nichts ausgesagt werden.

Zum Beispiel kann man zeigen, dass die VC–Dimension von Hyperebenen im $\mathbf{R^n}$ genau $n+1$ ist ([Vapnik, 1995], Kapitel 3.6). In Abbildung 2.2 können die drei Punkte in (a) in alle 8 verschiedenen Teilmengen aufgeteilt werden, man kann aber zeigen, dass alle Mengen von vier Punkte sich auf die Fälle (b) und (c) zurückführen lassen, in denen sich die markierten Punkte durch keine Hyperebene von den anderen trennen lassen.

Die VCdim reellwertiger Funktionen

Um das Konzept der Vapnik–Chervonenkis–Dimension auf reellwertige Funktionen zu übertragen, führt man reellwertige Funktionen auf Indikatorfunktionen zurück ([Vapnik, 1998], Kapitel 5.2). Zunächst beschränkt man sich auf gleichmäßig beschränkte Klassen von Funktionen $Q(z,\alpha), \alpha \in \Lambda$, d.h. es muss ein B existieren, so dass für alle $\alpha \in \Lambda$ gilt $|Q(z,\alpha)| \leq B$. Man betrachtet dann die Menge der Indikatorfunktionen $1_{\alpha,\beta}$ der Funktion $Q(z,\alpha)$.

$$1_{\alpha,\beta}(z) := \theta(Q(z,\alpha^*) - \beta), \quad \beta \in (inf_z Q(z,\alpha^*), sup_z Q(z,\alpha^*)).$$

$1_{\alpha,\beta}(z)$ gibt an, wann die Funktion $Q(z,\alpha)$ den Wert β übersteigt (siehe Abbildung 2.3).

Man beachte, dass $Q(z,\alpha^*)$ durch die Menge seiner Indikatorfunktionen eindeutig bestimmt ist, es ist nämlich

$$Q(z,\alpha^*) = sup\{\beta \in (inf_z Q(z,\alpha^*), sup_z Q(z,\alpha^*)) | \, \theta(Q(z,\alpha^*) - \beta) = 1\}$$

Die VC–Dimension der reellwertigen Funktionen $Q(z,\alpha), \alpha \in \Lambda$ ist die VC–Dimension der Indikatorfunktionen $\theta(Q(z,\alpha^*) - \beta), \alpha \in \Lambda, \beta \in \left[\inf_{\alpha,z} Q(z,\alpha), \sup_{\alpha,z} Q(z,\alpha)\right]$.

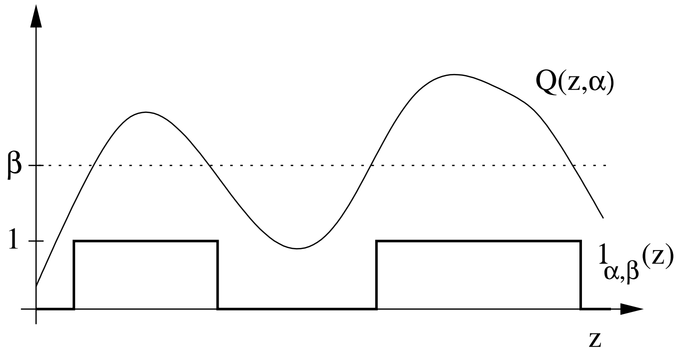


Abbildung 2.3: Indikatorfunktion $1_{\alpha,\beta}(z)$ zu $Q(z,\alpha)$.

2.2.2 Eine obere Schranke für das erwartete Risiko

Eins der Hauptergebnisse in [Vapnik, 1998] (Kapitel 5.10) ist die Gültigkeit einer oberen Schranke für das erwartete Risiko aufgrund des empirischen Risikos:

Theorem: Gegeben sei eine Menge von Funktionen $Q(z,\alpha), \alpha \in \Lambda$ mit endlicher VC–Dimension h. Existiert eine Zahl B, so dass $\forall \alpha \in \Lambda : 0 \leq Q(z,\alpha) \leq B$ gilt, dann gilt mit Wahrscheinlichkeit $1-\eta$:

$$R(\alpha_l) \leq R_{emp}(\alpha_l) + \frac{B\mathcal{E}(l)}{2}\left(1+\sqrt{1+\frac{4R_{emp}(\alpha)}{B\mathcal{E}(l)}}\right)$$

Dabei ist

$$\mathcal{E}(l) = 4\frac{h\left(\ln\frac{2l}{h}+1\right)-\ln\frac{\eta}{4}}{l}$$

Man schätzt das erwartete Risiko also durch eine Summe aus dem empirischen Risiko und einem Term ab, der über $\mathcal{E}(l)$ abhängig von der Kapazität des Lernalgorithmus ist (siehe Abbildung 2.4). Der zweite Summand sinkt mit steigendem l, so dass die Schranke für große Beispielmengen hauptsächlich aus dem empirischen Risiko gegeben ist. Dies erklärt, warum die empirische Risikominimierung für große Beispielmengen gute Ergebnisse liefert. Ist die Beispielmenge aber klein (im Verhältnis zu h), so wirkt sich der zweite Term stark aus. Hier muss also bei der Minimierung des Risikos zwischen empirischen Risiko und Kapazität des Lernprozesses abgewägt werden.

2.2.3 Das Prinzip der strukturellen Risikominimierung

Das Prinzip der strukturellen Risikominimierung ([Vapnik, 1998], Kapitel 6.1) beruht darauf, die VC–Dimension der benutzten Funktionenklasse als weiteren Parameter für den Lernalgorithmus zu benutzen. Dazu wird die Klasse S der Funktionen $Q(z,\alpha), \alpha \in \Lambda$ in Teilmengen $S_1 \subset S_2 \subset \ldots S_k \ldots$ aufgeteilt ($S_k = \{Q(z,\alpha), \alpha \in \Lambda_k\}$), so dass gilt

1. $S = \bigcup_k S_k$
2. Jedes S_k hat endliche VC–dimension h_k
3. Für alle k existiert ein B_k, so dass für alle $\alpha \in \Lambda_k$ gilt $0 \leq Q(z,\alpha) \leq B_k$. .

Diese Bedingungen stellen sicher, dass dass auf jeder Menge S_k die obere Schranke aus Abschnitt 2.2.2 gilt. Dabei müssen im Gegensatz zu den Funktionen aus jedem S_k die Funktionen aus ganz S weder endliche VC–Dimension haben, noch gleichmäßig beschränkt sein. Das Prinzip der strukturellen Risikominimierung kann damit wie folgt beschrieben werden:

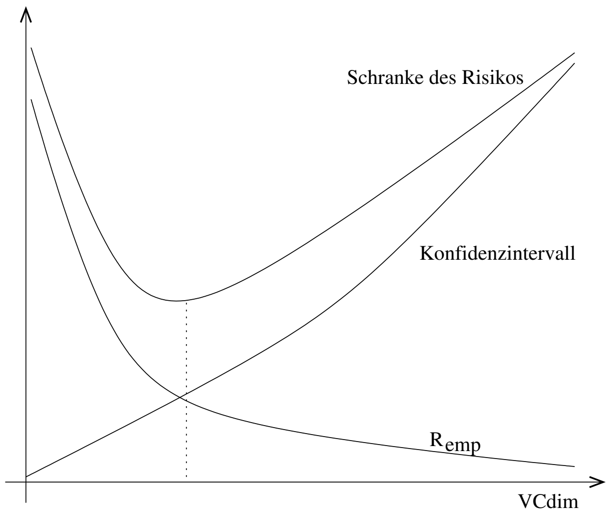


Abbildung 2.4: Obere Schranke des erwarteten Risikos.

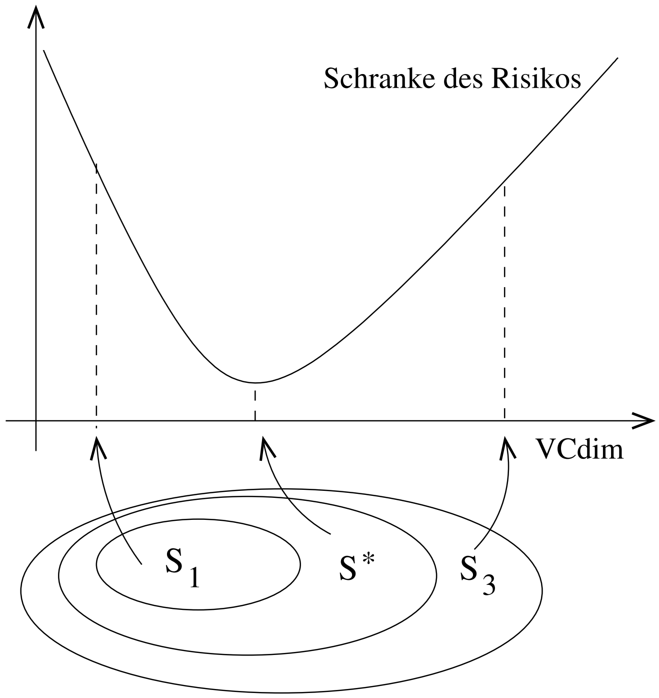


Abbildung 2.5: Das Prinzip der strukturellen Risikominimierung

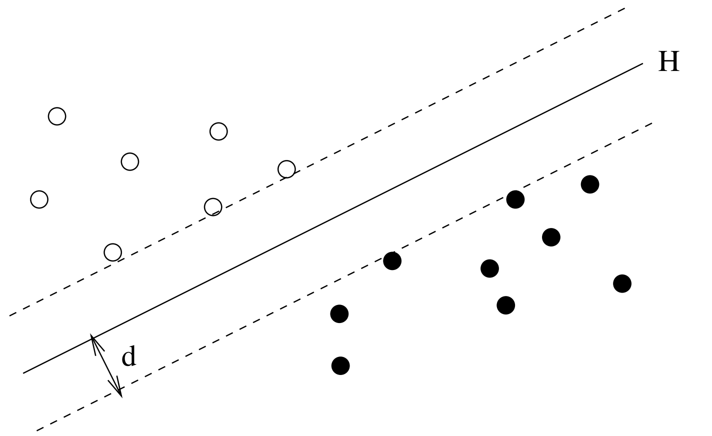


Abbildung 2.6: Optimale Hyperebene im $\mathbf{R^2}$.

Prinzip der strukturellen Risikominimierung: Bilde eine Struktur wie oben auf der Menge der Funktionen und minimiere das empirische Risiko schrittweise auf den Mengen $S_1, S_2, \ldots$. Wähle die Funktion aus der Menge S^*, die die obere Schranke aus 2.2.2 minimiert.

Es kann gezeigt werden, dass die strukturelle Risikominimierung konsistent ist ([Vapnik, 1998], Kapitel 6.3).

2.3 Support Vector Machines

Support Vector Machines implementieren das Prinzip der strukturellen Risikominimierung. Ganz allgemein bearbeiten sie die Aufgabe, aus Beispiele mit reellwertigen Attributen und $\{0, 1\}$- bzw. reellwertigen Klassifikationen (Mustererkennung bzw. Regression) lineare Funktionen zu lernen. Das Problem der strukturellen Risikominimierung wird dabei in ein quadratisches Optimierungsproblem übersetzt.

2.3.1 Support Vector Machines für Mustererkennung

Eine Menge von Beispielen $(x_1, y_1), \ldots, (x_n, y_n)$ mit $x_i \in \mathbf{R}$ und $y_i \in \{-1, +1\}$ heißt linear trennbar, wenn eine Hyperebene H,

$$H = \{x | \Phi \cdot x + b = 0\}$$

mit $||\Phi|| = 1$ existiert, so dass $\Phi \cdot x_i > b$ falls $y_i > 0$ und $\Phi \cdot x_i < b$ falls $y_i < 0$ gilt (Zusammengefasst: $y_i(\Phi \cdot x_i - b) > 0$). Eine trennende Hyperebene heißt optimale Hyperebene, wenn Abstand d zu den Beispielen maximal ist (siehe Abbildung 2.6).

Es kann gezeigt werden ([Vapnik, 1998], Kapitel 10.1), dass die optimale Hyperebene eindeutig bestimmt ist. Weiter ist das Finden einer optimalen Hyperebene äquivalent zum Finden eines Vektors w und einer Konstanten b_0 ist, so dass gilt

$$y_i(w \cdot x_i + b_0) \geq 1$$

und der Vector w minimale Norm $||w|| = w \cdot w$ hat.

Ein Grund für die Effizienz der Support Vector Machine ist, dass nur ein Teil der Beispiele für die Konstruktion der optimalen Hyperebene benötigt werden. Diese Beispiele heißen Support Vektoren. Lässt man alle anderen Beispiele weg, so erhält man trotzdem wieder dieselbe optimale Hyperebene (Abbildung 2.7).

Intuitiv sollte die Benutzung der optimalen Hyperebene zu einer guten Generalisierung führen, da so die Beispiele auch noch richtig klassifiziert werden, wenn man sie ein klein wenig verschiebt

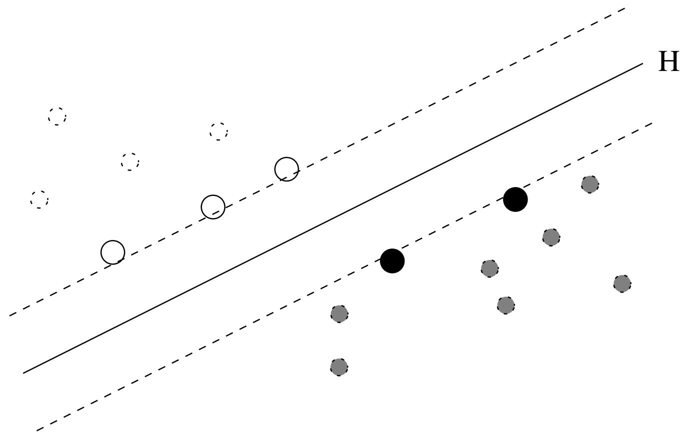


Abbildung 2.7: Weglassen der nicht-Supportvektoren ändert die optimale Hyperebene nicht.

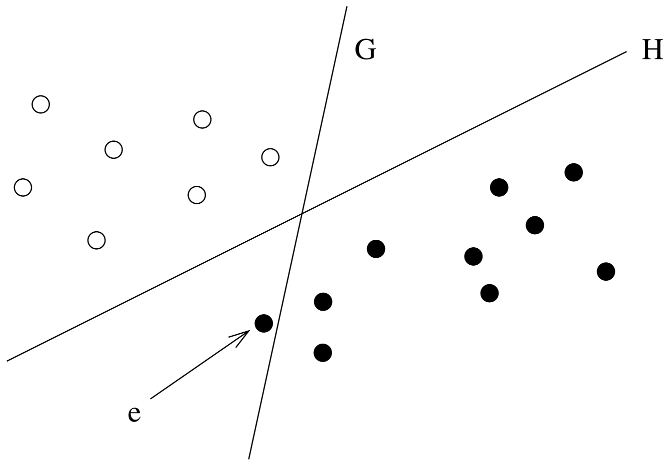


Abbildung 2.8: Das neue Beispiel e wird von der optimalen Hyperebene H noch richtig klassifiziert, von der trennenden Hyperebene G aber nicht.

(siehe Abbildung 2.8). Das Prinzip der Support Vector Machine ist also gleichzeitig das empirische Risiko zu minimieren (die Beispiele zu trennen) und möglichst gut zu generalisieren. Hier wird bereits ein Zusammenhang mit der strukturellen Risikominimierung deutlich, der im nächsten Abschnitt präzisiert wird.

Support Vector Machines und strukturelle Risikominimierung

Der Zusammenhang zwischen Support Vector Machines und der strukturellen Risikominimierung besteht in folgendem Theorem. Für eine Menge $(x_1, \ldots, x_n)$ von Punkten heißt eine Hyperebene

$$H = \{x | w \cdot x + b = 0\}$$

kanonische Hyperebene, wenn gilt

$$\inf_{i=1,\ldots,n} |x_i \cdot w + b| = 1.$$

Das bedeutet lediglich, dass die Hyperebene in Bezug auf die Punkte $(x_1, \ldots, x_n)$ normalisiert wird. Über die VC–Dimension von kanonischen Hyperebenen ist nun folgendes bekannt ([Vapnik, 1998], Kapitel 10.3):

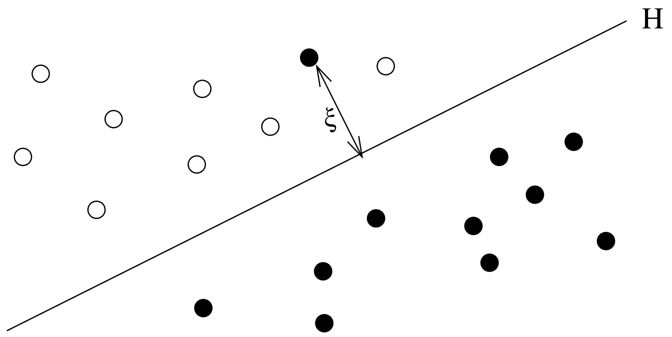


Abbildung 2.9: Das fehlklassifizierte Beispiel besitzt eine Abweichung $\xi > 0$.

VC–Dimension von kanonischen Hyperebenen: Eine Menge von kanonischen Hyperebenen $H = \{x | w \cdot x + b = 0\}$ mit $||w|| < A$ hat auf einer Menge von Beispielen X mit $|x| < D$ für alle $x \in X$ eine VC–Dimension h von höchstens

$$h \leq \min\left(\left[D^2 A^2\right], n\right) + 1.$$

Da das Maximieren des Abstands in der SVM dem Minimieren von $||w||$ (also auch von A) entspricht, entspricht das Prinzip der Support Vector Machine also dem Minimieren der VC–Dimension.

Der allgemeine Fall

Im Allgemeinen kann man nicht davon ausgehen, dass die Daten tatsächlich linear trennbar sind. In diesem Fall muss man Fehlklassifikation hinnehmen, etwa indem man einige Beispiele aus der Trainingsmenge herausnimmt. In [Cortes und Vapnik, 1995] werden dazu neue Variablen $\xi_i \geq 0$ eingeführt, die dem Fehler der Klassifikation auf dem Beispiel i entsprechen (siehe Abbildung 2.9) und die Bedingungen

$$y_i(w \cdot x_i + b) \geq 1$$

ersetzt durch

$$y_i(w \cdot x_i + b) \geq 1 - \xi_i.$$

Um neben $||w||$ auch die Abweichungen zu minimieren, wird für eine Konstante C und eine monotone, konvexe Funktion F das Funktional

$$\frac{1}{2}w^2 + CF\left(\sum_{i=1}^{n} \xi_i^\sigma\right)$$

minimiert. Wählt man σ klein genug, so entspricht die Minimierung des Funktionals dem Entfernen einer minimalen Anzahl von Beispielen und dem Bilden der optimalen Hyperebene für den Rest. Man führt hier also eine Verlustfunktion ein, die angibt wie schwer der Fehler ξ_i auf einem Beispiel x_i bestraft wird.

Man erhält ein möglichst einfach zu berechnendes Problem, wenn man $\sigma = 1$ oder $\sigma = 2$ wählt. Im einfachsten Fall ist $F(x) = x$ und $\sigma = 1$. Die zu lösende Aufgabe ist also:

Optimierungsaufgabe der SVM: Minimiere

$$\Phi(w, \xi) = \frac{1}{2}w^2 + C\sum_{i=1}^{n} \xi_i$$

unter den Nebenbedingungen

$$y_i(w \cdot x_i + b) \geq 1 - \xi_i, i = 1, \ldots, n$$

und

$$\xi_i \geq 0, i = 1, \ldots, n.$$

Eine Optimierungsaufgabe dieser Art heißt quadratisches Optimierungsproblem. Es ist bekannt, dass man, um dieses Optimierungsproblem zu lösen, den Sattelpunkt des Lagrangefunktionals

$$L(w, \xi, b, \Lambda, \Gamma) = \frac{1}{2} w \cdot w + C \sum_{i=1}^{n} \xi_i - \sum_{i=1}^{n} \alpha_i \left(y_i(w \cdot x_i + b) - 1 + \xi_i \right) - \sum_{i=1}^{n} \gamma_i \xi_i$$

finden muss (siehe etwa [Grossmann und Terno, 1997]). Das heißt, man muss das Minimum von L bezüglich w, b und ξ_i und das Maximum bezüglich dualen Variablen α_i und γ_i mit $\alpha_i, \gamma_i \geq 0$ finden. Weiter müssen im Extrempunkt die partiellen Ableitungen von L bezüglich w, b und ξ_i gleich Null sein. Daraus erhält man

$$\frac{\partial L}{\partial w} = w - \sum_{i=1}^{n} \alpha_i y_i x_i \stackrel{!}{=} 0$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{n} \alpha_i y_i \stackrel{!}{=} 0$$

$$\frac{\partial L}{\partial \xi_i} = C - \alpha_i - \gamma_i$$

.

Also gilt

$$w = \sum_{i=1}^{n} \alpha_i y_i x_i$$

$$\sum_{i=1}^{n} \alpha_i y_i = 0$$

$$C = \alpha_i + \gamma_i$$

Setzt man diese Gleichungen wieder in die Definition des Lagrange-Funktionals ein, so erhält man

$$W(\alpha) = -\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_i \cdot x_j + \sum_{i=1}^{n} \alpha_i$$

Dieser Ausdruck hängt nur noch von α ab. Um den gesuchten Sattelpunkt zu finden muss also $W(\alpha)$ maximiert werden unter den Nebenbedingungen

$$\sum_{i=1}^{n} \alpha_i y_i = 0$$

$$0 \leq \alpha_i \leq C$$

Die letzte Nebenbedingung folgt durch Vernachlässigen von γ_i, das nur in einer Nebenbedingung auftaucht. Damit wird das Finden der optimalen Hyperebene auf eine konvexen, quadratische Minimierungsaufgabe mit der positiv semidefiniten Matrix $K = (y_i y_j x_i \cdot x_j)_{1 \leq i,j \leq n}$ zurückgeführt.

Die gelernte Entscheidungsfunktion hat damit die Form

$$\begin{aligned} F(x) &= \operatorname{sign}(w \cdot x - b) && (2.3) \\ &= \operatorname{sign}\left(\sum_{i=1}^{n} \alpha_i y_i x_i \cdot x - b \right) && (2.4) \end{aligned}$$

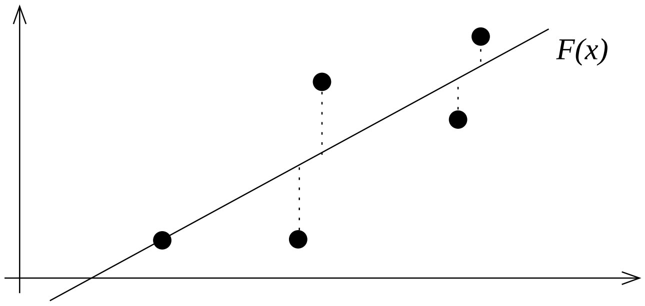


Abbildung 2.10: Die Funktion $F(x)$ minimiert die Summe der Abstände zu den Beispielen.

wobei die Konstante b noch aus den Trainingsbeispielen bestimmt werden muss. Insbesondere sieht man, dass Beispiele (x_i, y_i) mit Lagrange-Multiplikatoren $\alpha_i = 0$ nichts zur Entscheidungsfunktion beitragen. Die x_i mit nicht verschwindenden Lagrange-Multiplikatoren nennt man Support-Vektoren. Es gilt also (mit $SV := \{i|\alpha_i \neq 0\}$):

$$F(x) \quad = \quad \mathrm{sign}\left(\sum_{i \in SV} \alpha_i y_i x_i \cdot x - b\right) \tag{2.5}$$

Die Konstante b kann mittels der Kuhn-Tucker-Bedingungen ([Grossmann und Terno, 1997]) bestimmt werden. Diese besagen, dass im Lösungspunkt der quadratischen Aufgaben das Produkt der Nebenbedingung mit ihrer dualen Variable verschwindet.

$$\alpha_i \cdot (y_i(w \cdot x_i + b) - 1 + \xi_i) = 0 \tag{2.6}$$

Man kann also aus jedem Support-Vektor x_i die fehlende Konstante b berechnen als

$$b \quad = \quad \frac{1 - \xi_i}{y_i} - w \cdot x_i \tag{2.7}$$

Um numerische Ungenauigkeiten auszugleichen ist es unter Umständen günstiger, b als Durchschnitt der jeweiligen Werte aus allen Support-Vektoren auszurechnen.

$$b \quad = \quad \frac{1}{|SV|} \sum_{i \in SV} \left(\frac{1 - \xi_i}{y_i} - w \cdot x_i\right) \tag{2.8}$$

2.3.2 Support Vector Machines für Regression

Das Konzept der Support Vector Machine kann erweitert werden um reellwertige Funktionen zu lernen. Die grundlegende Idee dabei ist Werte $y_i \in \mathbf{R}$ zu erlauben und die Funktionswerte mit einer linearen Funktion zu approximieren.

Im Fall der Mustererkennung war das Ziel, die Daten linear zu trennen und dabei möglichst wenige der Beispiele falsch zu klassifizieren. Stattdessen wird jetzt eine Verlustfunktion $L(y, f(x, \alpha))$ vorgegeben und das Ziel ist, den erwarteten Verlust der gelernten Funktion $F(x) = w \cdot x - b$ über alle Trainingsbeispiele zu minimieren (Abbildung 2.10).

Oft wählt man eine lineare, ε-insensitive Verlustfunktion (Abbildung 2.11), d.h. für festes $\varepsilon \geq 0$ ist

$$L(y, f(x, \alpha) = \begin{cases} 0 & \text{falls } |y - f(x, \alpha)| \leq \varepsilon \\ |y - f(x, \alpha)| - \varepsilon & \text{sonst} \end{cases}$$

oder eine quadratische ε-insensitive Verlustfunktion

$$L(y, f(x, \alpha) = \begin{cases} 0 & \text{falls } |y - f(x, \alpha)| \leq \varepsilon \\ (|y - f(x, \alpha)| - \varepsilon)^2 & \text{sonst} \end{cases}$$

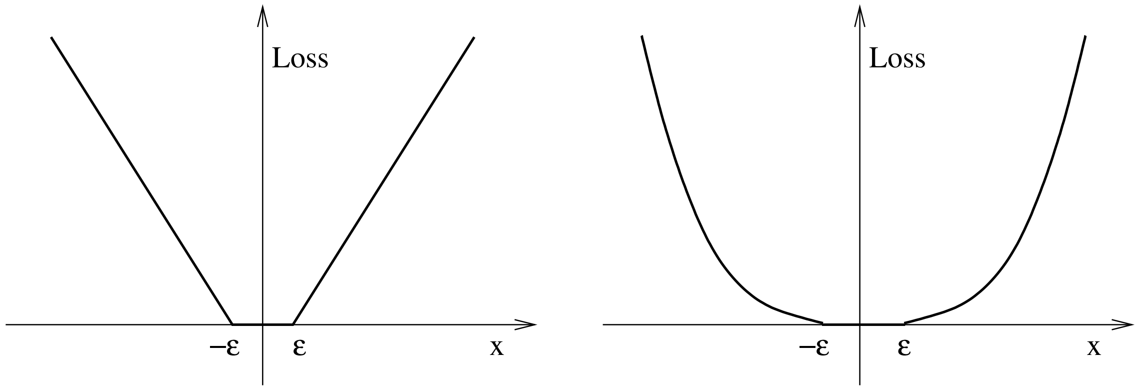


Abbildung 2.11: Lineare und quadratische Verlustfunktion.

Wieder führt man neue Variablen ein, um den Fehler auf den Trainingsbeispielen darzustellen. Diesmal muss man aber je eine Variable ξ_i für die positive und x_i^* für die negative Abweichung auf Beispiel x_i einführen.

Wieder lässt sich das Problem in ein quadratisches Minimierungsproblem umformen.

Lineare Kostenfunktion

Die Zielfunktion ist für einen gegebenen Wert $C \in \mathbf{R}_{>0}$ gegeben durch

$$\Phi(w, \xi, \xi^*) = \frac{1}{2}(w^T w) + C\left(\sum_{i=1}^{l} \xi^* + \sum_{i=1}^{l} \xi\right)$$

Φ soll minimiert werden unter den Nebenbedingungen

$$\begin{aligned} y_i - (w^T x_i) - b &\leq \varepsilon + \xi_i^* \quad , i = 1, \ldots, n \quad \text{(negative Abweichung}^1\text{)} && (2.9)\\ (w^T x_i) + b - y_i &\leq \varepsilon + \xi_i \quad , i = 1, \ldots, n \quad \text{(positive Abweichung)} && (2.10)\\ \xi_i^* &\geq 0 \quad , i = 1, \ldots, n && (2.11)\\ \xi_i^* &\geq 0 \quad , i = 1, \ldots, n && (2.12) \end{aligned}$$

Die Minimierung von Φ ist äquivalent zur Bestimmung eines Sattelpunkts der Lagrangefunktion $L(w, b, \xi, \xi^*; \alpha, \alpha^*, \gamma, \gamma^*)$ (Minimierung bzgl. w, b, ξ, ξ^* und Maximierung bzgl. der Lagrange-Multiplikatoren $\alpha, \alpha^*, \gamma, \gamma^*$) mit $\alpha, \alpha^*, \gamma, \gamma^* \geq 0$.

$$\begin{aligned} L(w, b, \xi, \xi^*; \alpha, \alpha^*, \gamma, \gamma^*) &= \frac{1}{2} w^T w + C\left(\sum_{i=1}^{l} \xi_i^* + \sum_{i=1}^{l} \xi_i\right) - \sum_{i=1}^{l} \alpha_i^* \left(w^T x_i - y_i + b + \varepsilon + \xi_i^*\right)\\ &\quad - \sum_{i=1}^{l} \alpha_i \left(y_i - w^T x_i - b + \varepsilon + \xi_i^*\right) - \sum_{i=1}^{l} \gamma_i \xi_i - \sum_{i=1}^{l} \gamma_i^* \xi_i^* \end{aligned}$$

Daraus erhält man, dass im Sattelpunkt folgende Bedingungen erfüllt sein müssen

$$\begin{aligned} w &\stackrel{!}{=} \sum_{i=1}^{l} (\alpha_i^* - \alpha_i) x_i && (2.13)\\ \sum_{i=0}^{l} \alpha_i &\stackrel{!}{=} \sum_{i=0}^{l} \alpha_i^* && (2.14)\\ C &\stackrel{!}{=} \alpha_i + \gamma_i \quad , i = 1, \cdots, l && (2.15)\\ C &\stackrel{!}{=} \alpha_i^* + \gamma_i^* \quad , i = 1, \cdots, l && (2.16) \end{aligned}$$

Einsetzen der Bedingungen liefert dann

$$\begin{aligned} L(w,b,\xi,\xi^*;\alpha,\alpha^*,\gamma,\gamma^*) &= W(\alpha,\alpha^*) && (2.17)\\ &= -\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i^*-\alpha_i)(\alpha_j^*-\alpha_j)(x_i\cdot x_j)\\ &\quad +\sum_{i=1}^{l} y_i(\alpha_i^*-\alpha_i)-\sum_{i=1}^{l}\varepsilon(\alpha_i^*+\alpha_i) && (2.18)\end{aligned}$$

Es ist also $W(\alpha,\alpha^*)$ zu maximieren unter den Nebenbedingungen

$$\begin{aligned} 0\le\alpha_i,\alpha_i^*\le C & \qquad ,i=1,\dots,n && (2.19)\\ \sum_{i=1}^{l}\alpha_i^* &= \sum_{i=1}^{l}\alpha_i && (2.20)\end{aligned}$$

Quadratische Kostenfunktion

Die Zielfunktion ist für einen gegebenen Wert $C\in\mathbf{R}_{>0}$ gegeben durch

$$\Phi(w,\xi,\xi^*)=\frac{1}{2}(w^Tw)+C\left(\sum_{i=1}^{n}\xi^{*2}+\sum_{i=1}^{n}\xi^2\right)$$

Φ soll minimiert werden unter den Nebenbedingungen

$$\begin{aligned} y_i-(w^Tx_i)-b &\le \varepsilon+\xi_i^* \quad ,i=1,\dots,n \quad \text{(negative Abweichung)} && (2.21)\\ (w^Tx_i)+b-y_i &\le \varepsilon+\xi_i \quad ,i=1,\dots,n \quad \text{(positive Abweichung)} && (2.22)\\ \xi_i^* &\ge 0 \quad ,i=1,\dots,n && (2.23)\\ \xi_i &\ge 0 \quad ,i=1,\dots,n && (2.24)\end{aligned}$$

die Minimierung von Φ äquivalent zur Bestimmung eines Sattelpunkts der Lagrangefunktion $L(w,b,\xi,\xi^*;\alpha,\alpha^*,\gamma,\gamma^*)$ (Minimierung bzgl. w,b,ξ,ξ^* und Maximierung bzgl. der Lagrange-Multiplikatoren $\alpha,\alpha^*,\gamma,\gamma^*$) mit $\alpha,\alpha^*,\gamma,\gamma^*\ge 0$.

$$\begin{aligned} L(w,b,\xi,\xi^*;\alpha,\alpha^*,\gamma,\gamma^*) &= \frac{1}{2}w^Tw+\frac{1}{2}C\left(\sum_{i=1}^{n}\xi_i^{*2}+\sum_{i=1}^{n}\xi_i^2\right)-\sum_{i=1}^{n}\alpha_i^*\left(w^Tx_i-y_i+b+\varepsilon+\xi_i^*\right)\\ &\quad -\sum_{i=1}^{n}\alpha_i\left(y_i-w^Tx_i-b+\varepsilon+\xi_i^*\right)-\sum_{i=1}^{n}\gamma_i\xi_i-\sum_{i=1}^{n}\gamma_i^*\xi_i^*\end{aligned}$$

Anwendung der Kuhn-Tucker-Bedingungen ergibt, dass im Sattelpunkt folgende Bedingungen erfüllt sein müssen

$$\begin{aligned} w &\stackrel{!}{=} \sum_{i=1}^{n}(\alpha_i^*-\alpha_i)x_i && (2.25)\\ \sum_{i=0}^{n}\alpha_i &\stackrel{!}{=} \sum_{i=0}^{l}\alpha_i^* && (2.26)\\ C\xi_i &\stackrel{!}{=} \alpha_i+\gamma_i \quad ,i=1,\cdots,l && (2.27)\\ C\xi_i^* &\stackrel{!}{=} \alpha_i^*+\gamma_i^* \quad ,i=1,\cdots,l && (2.28)\end{aligned}$$

Einsetzen der Bedingungen liefert dann

$$
\begin{aligned}
L(w,b,\xi,\xi^*;\alpha,\alpha^*,\gamma,\gamma^*) &= W(\alpha,\alpha^*) && (2.29)\\
&= -\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i^*-\alpha_i)(\alpha_j^*-\alpha_j)(x_i\cdot x_j)\\
&\quad +\sum_{i=1}^{n} y_i(\alpha_i^*-\alpha_i)-\sum_{i=1}^{n}\varepsilon(\alpha_i^*+\alpha_i)-\frac{1}{C}\sum_{i=1}^{n}(\alpha_i^{*2}+\alpha_i^2) && (2.30)\\
&\quad -\frac{1}{C}\sum_{i=1}^{n} 2\alpha_i\gamma_i+\gamma_i^2+2\alpha_i^*\gamma_i^*+\gamma_i^{*2} && (2.31)\\
&= -\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i^*-\alpha_i)(\alpha_j^*-\alpha_j)(x_i\cdot x_j)\\
&\quad +\sum_{i=1}^{n} y_i(\alpha_i^*-\alpha_i)-\sum_{i=1}^{n}\varepsilon(\alpha_i^*+\alpha_i)-\frac{1}{C}\sum_{i=1}^{n}(\alpha_i^{*2}+\alpha_i^2) && (2.32)
\end{aligned}
$$

Die letzte Gleichheit folgt dabei, da die letzte Summe immer kleiner oder gleich Null ist ($\alpha,\alpha^*,\gamma,\gamma^*\geq 0$) und W maximiert werden soll, d.h. man kann die γ,γ^* weglassen ohne die Lösung des Problems zu verändern. Es ist also $W(\alpha,\alpha^*)$ zu maximieren unter den Nebenbedingungen

$$
\begin{aligned}
0\leq \alpha_i,\alpha_i^* & \quad ,i=1,\ldots,n && (2.33)\\
\sum_{i=1}^{n}\alpha_i^* &= \sum_{i=1}^{n}\alpha_i && (2.34)
\end{aligned}
$$

Der Unterschied zwischen der linearen und der quadratischen Verlustfunktion ist also die fehlende obere Schranke der α_i und α_i^* sowie der zusätzliche Term $\frac{1}{C}$ vor den α_i^2,α_i^{*2}.

2.3.3 Kernelfunktionen

In der bisherigen Formulierung können die Support Vector Machines nur lineare Funktionen lernen. Um auch nichtlineare Funktionen lernen zu können benutzt man folgende Idee: Man transformiert die Daten mit einer Funktion $\Phi: X\to\mathcal{X}$ aus dem Eingaberaum X in einen Featureraum $\mathcal{X}$ und lernt dort eine lineare Funktion (Abbildung 2.12). Damit hat die gelernte Funktion die (nichtlineare) Form

$$F(x)=\sum_{i=1}^{n}\beta_i\Phi(x_i)\cdot\Phi(x)-b$$

Betrachtet man den Algorithmus der Support Vector Machine so fällt auf, dass die genaue Form der Abbildung Φ nicht benötigt wird, es genügt für je zwei Punkte $x,y\in X$ den Wert des Skalarprodukts $\Phi(x)\cdot\Phi(y)$ zu kennen. Die entscheidenden Idee ([Boser et al., 1992]) ist nun, dass es genügt eine Kernelfunktion k zu kennen, so dass eine Abbildung $\Phi: X\to\mathcal{X}$ in einen geeigneten Raum $\mathcal{X}$ existiert, die die Gleichung

$$\forall x,y\in X: k(x,y)=\Phi(x)\cdot\Phi(y)$$

erfüllt.

Um zu entscheiden, ob für eine Funktion k solch ein Φ existiert, kann man den aus der Funktionalanalysis bekannten Satz von Mercer verwenden.

Satz von Mercer: Eine stetige, symmetrische Funktion $K(u,v)$ in $L_2(C)$ hat eine Darstellung der Form

$$K(u,v)=\sum_{k=0}^{\infty}a_k z_k(u)z_k(v) \tag{2.35}$$

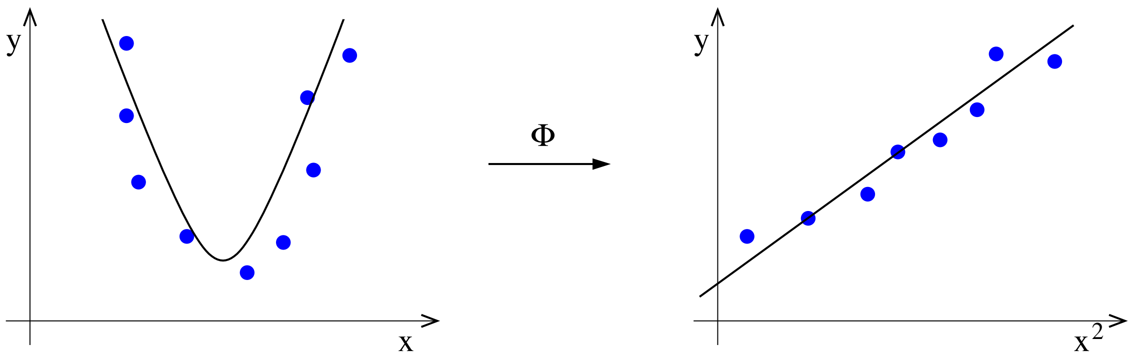


Abbildung 2.12: Abbildung vom Eingaberaum (x, y) in den Featureraum (x^2, y).

mit Koeffizienten $a_k > 0$ genau dann, wenn für alle $g \in L_2(C)$ gilt:

$$\int_C \int_C K(u,v)g(u)g(v)dudv \geq 0$$

C sei dabei eine kompakte Teilmenge des $\mathbf{R^n}$. Eine Kernelfunktion, die diese Bedingung erfüllt heißt auch Mercer-Kernel.

Gleichung 2.35 besagt gerade, dass ein Featureraum existiert, in dem sich K durch ein Skalarprodukt ausdrücken lässt. Beispiele solcher Kernelfunktionen sind:

- Polynome des Grads d: $K(x,y) = [(x \cdot y) + 1]^d$, $d \in N$
- Splines: $K(x,y) = \sum_{r=0}^{d} x_r y_r + \sum_{i=1}^{m} (x - t_i)_+^d (y - t_i)_+^d$
- Radiale Basisfunktionen: $K_\gamma(x,y) = \exp(-\gamma |x - y|^2)$, $\gamma \in Re_{\geq 0}$
- Zweilagige Neuronale Netze: $K(x,y) = S(v(x \cdot y) + c)$, S sigmoide Funktion, $v, c \in Re$ (Nur für bestimmte $v, c \in \mathbf{R}$)
- Anova-Kernel: $K(x,y) = \exp(-\gamma |x - y|^2)^n$, $\gamma \in Re_{\geq 0}$, $n \in \mathbf{N}$

Der hier vorgestellte Anova-Kernel ist nur ein Spezialfall der Anova-Kernels für die radiale Basisfunktion. Die Idee der Anova-Kernel ist, für einen Kernel, der als Produkt eindimensionaler Kernel geschrieben werden kann, den Kernel in Teile zu zerlegen, die jeweils nur von einer Teilmenge der Variablen abhängen. Eine Untersuchung der Support Vector Regressionschätzung mit Anova-Kerneln findet sich in [Stitson et al., 1997].

2.3.4 Praktische Umsetzung

In der Praxis ist von wesentlicher Bedeutung das quadratische Optimierungsproblem möglichst effizient zu lösen. Dazu bieten sich die sogenannten *innere Punkt Methoden* an, mit denen ein globales Minimum der Zielfunktion in polynomieller Zeit gefunden werden kann. Ein weitere Vorteil dieser Methoden ist, dass man den Wert der Konstanten b in der Entscheidungsfunktion als Nebenergebnis aus dem Optimierer erhält (siehe [Smola, 1998]).

Eine effiziente Verwaltung der Beispiele hat ebenfalls einen großen Einfluß auf die Laufzeit. Es empfiehlt sich, alle Beispiele im Hauptspeicher vorrätig zu halten. Ist dies nicht möglich, kann man das Optimierungsproblem in mehrere kleine Optimierungsprobleme zerlegen ([Osuna et al., 1997], [Joachims, 1999]). Dazu hält man die Werte der Lagrange-Multiplikatoren einigen Stellen fest und löst nur das durch die restlichen Variablen gegebene Problem. Nach jedem Schritt tauscht man dann einen Teil der festgesetzten Werte wieder aus.

Haben die verwendeten Attribute deutlich unterschiedliche Größenordnungen, so kann dies zu Problemen mit der numerischen Genauigkeit führen. Dann ist es sinnvoll die Werte linear zu skalieren, so dass alle Werte im Gleichen Intervall liegen oder gleichen Erwartungswert und Varianz haben.

Eine Zusammenfassung und weiterführende Gedanken zur Regressionsschätzung mit Support Vector Machines finden sich unter anderem in [Smola et al., 1996] und in [Smola und Schölkopf, 1998].

Kapitel 3

Probleme der Zeitreihenprognose in Warenwirtschaftssystemen

Ein wichtiges Teilgebiet in der Betriebswirtschaftslehre ist die Logistik. Laut der Definition in [Günther und Tempelmeier, 1997] bezeichnet die Logistik in der Betriebswirtschaftslehre „eine Querschnittsfunktion, deren Aufgabe es ist, räumliche, zeitliche und mengenmäßige Differenzen zwischen Angebot und Nachfrage zu überbrücken“. Dazu müssen der Transport, der Umschlag und die Lagerung der Waren geplant und koordiniert werden.

Die Schwierigkeit dieses Problem stammt daher, dass die relevanten Daten zufälligen Schwankungen unterworfen sind. Laut [Günther und Tempelmeier, 1997] gibt es vier Ursachen dieser Unsicherheit.

1. Die Nachfragemenge pro Periode ist nicht sicher.
2. Die Wiederbeschaffungszeit einer Lagerbestellung ist nicht sicher.
3. Die Lagerzugangsmenge weicht von der Bestellung ab.
4. Die Aufzeichnungen der Lagerbestandsführung stimmen nicht mit den tatsächlich vorhandenen Beständen überein.

Für den hier betrachteten Anwendungsfall des Einzelhandels ist vor allem die Unsicherheit in der Nachfragemenge entscheidend, da die Nachfrage nach den Produkten auf den einzelnen Kaufentscheidungen einer Vielzahl von Kunden beruhen, die von vielen externen Einflüssen abhängen. Im Vergleich zu diesen Schwankungen lassen sich die übrigen Unsicherheiten durch eine sorgfältige Lagerhaltung und gute Abstimmung mit den Lieferanten relativ einfach minimieren.

Der Prozess der Wiederbeschaffung von Waren läuft in drei Phasen ab (siehe Abbildung 3.1). Zuerst wird eine Bestellung über eine gewisse Menge von Artikeln aufgegeben. Dies kann z.B. in festen Abständen bestehen oder immer dann, wenn der Lagerbestand eine bestimmte Größe unterschreitet. Während der Lieferzeit der Artikel sinkt der Bestand durch die kontinuierliche Nachfrage weiter. Eventuell werden alle Artikel verbraucht, so dass die Nachfrage nicht mehr bearbeitet werden kann bis die Bestellung eintrifft und dadurch wieder genug Ware verfügbar ist. Eine optimale Lagerhaltungspolitik wäre so beschaffen, dass gerade zum Eintreffen der neuen Bestellung keine Waren mehr im Lager verfügbar sind, so dass sowohl Überkapazitäten als auch fehlende Bestände vermieden werden.

Tatsächlich ist es sogar unerheblich ob bei der Aufgabe einer neuen Bestellung die alte Bestellung schon eingetroffen ist, wenn man die bereits bestellte Ware bei der Bestimmung des neuen Bedarfs berücksichtigt. Wichtig ist lediglich, dass man den Bedarf zu jedem Zeitpunkt möglichst exakt vorhersagen kann, da man daraus für jeden Zeitpunkt den optimalen Lagerbestand und die nötigen Bestellungen berechnen kann. Umgekehrt bedeutet dies auch, dass für die Prognose des Bedarfs der jeweilige Lagerbestand und die Lagerhaltungspolitik unerheblich sind, da die nötigen

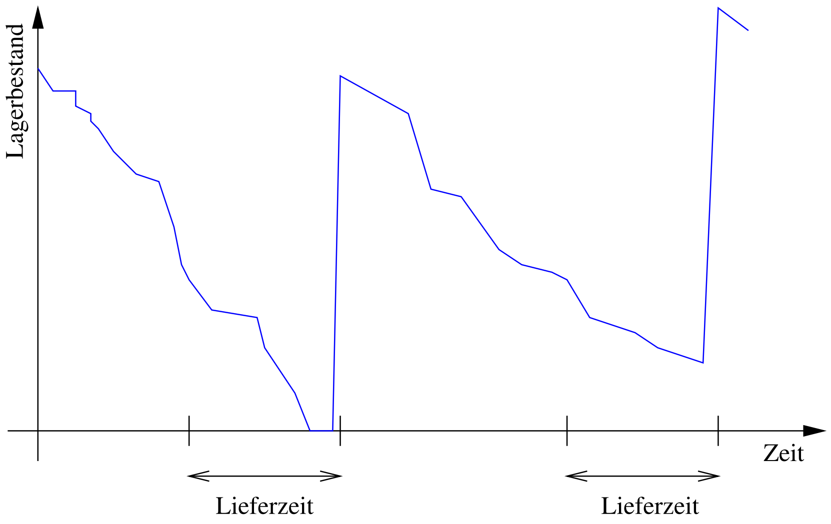


Abbildung 3.1: Bestellzyklus

Aktionen erst in einem zweiten Schritt nach der Prognose vom Warenwirtschaftssystem bestimmt werden.

Die Aufgabenstellung der Prognose wird durch sechs Teilprobleme charakterisiert (vergleiche [Arminger und Schneider, 1999]).

1. Asymmetrische Kosten von Fehlprognosen
2. Betriebswirtschaftliche Restriktionen
3. Beeinflussung der Verkaufszahlen durch globale Parameter
4. Beeinflussung der Verkaufszahlen durch artikelspezifische Parameter
5. Einflüsse innerhalb einer größeren Artikelgruppe
6. Kurze Zeitreihen

Im folgenden werde ich diese Teilprobleme jeweils beschreiben.

3.1 Asymmetrische Kosten von Fehlprognosen

Aus betriebswirtschaftlicher Sicht muss zwischen den Kosten für eine zu hohe und eine zu niedrige Prognose streng unterschieden werden. Ist der prognostizierte Bedarf höher als die tatsächlich eintretende Nachfrage, so müssen die überflüssigen Artikel für mindestens eine Verkaufsperiode gelagert werden, wodurch sie Kosten verursachen. Da der Restbestand bei der nächsten Bestellung berücksichtigt wird und dementsprechend weniger Artikel bestellt werden, kann man im Allgemeinen davon ausgehen, dass die Artikel genau eine Verkaufsperiode lang gelagert werden müssen. Diese Annahme setzt eine gewisse Genauigkeit der Prognose voraus.

Die überflüssigen Artikel verursachen dabei vor allem Zinskosten, die nach einem innerbetrieblichen Schema berechnet werden. Da die Zinskosten lediglich für eine Verkaufsperiode anfallen sind diese linear zu der Stückzahl der Artikel. Bei großen Abweichungen müssen allerdings auch andere Faktoren berücksichtigt werden. So ist zum Beispiel die Lagerkapazität begrenzt, d.h. bei Überschreitung einer maximalen Stückzahl müssen Regale im Lager umgeräumt werden, neue Lagerarbeiter eingestellt oder die Lagerkapazität erhöht werden. Dies verursacht sehr hohe Kosten, so dass es notwendig sein mag, hohe Abweichungen strenger zu bestrafen als kleine Abweichungen.

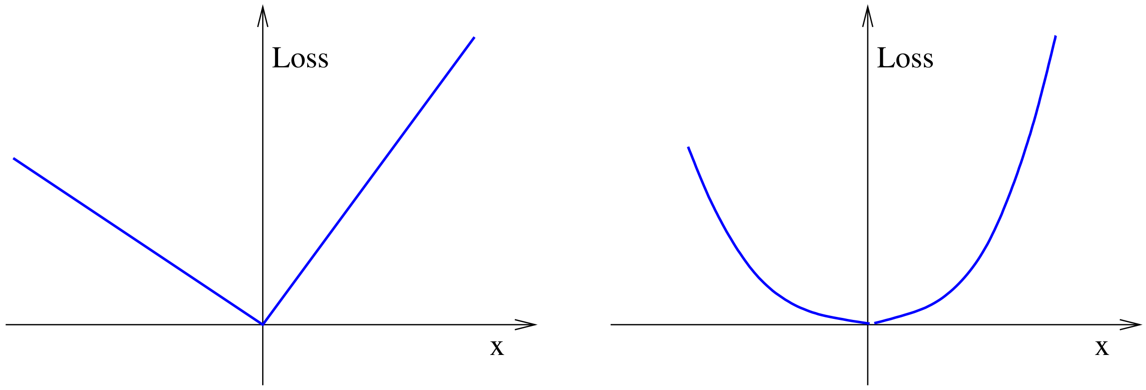


Abbildung 3.2: Asymmetrische lineare und quadratische Verlustfunktion.

Unterschätzt man die tatsächlich benötigte Stückzahl, so sind vor Eintreffen der nächsten Bestellung keine Artikel mehr im Geschäft verfügbar. In diesem Fall entgeht einem der Profit, den man erhalten hätte, wenn die Artikel tatsächlich verkauft worden wären. Dieser Profit ist im Einzelfall schlecht zu quantifizieren, es ist aber zumindest eine durchschnittliche Profitspanne des Unternehmens bekannt, so dass auch hier die Kosten als linear zur Stückzahl angesehen werden können. Noch problematischer ist jedoch, dass erfahrungsgemäß Kunden, die einen gewünschten Artikel nicht erhalten, zur Konkurrenz wechseln und ihren gesamten Einkauf dort vollziehen. Es tritt also ein Imageverlust des Geschäfts. Aufgrund dessen wird man die Kosten für das Unterschätzen des Bedarfs noch höher ansetzen, als durch den entgangenen Profit angezeigt wäre. Auch hier kann es also sinnvoll sein, höhere Abweichungen wesentlich stärker zu bestrafen.

In [Arminger und Götz, 1999] werden als geeignete Funktionen neben linearen asymmetrischen Kostenfunktionen auch quadratische asymmetrische Kostenfunktionen betrachtet (Abbildung 3.2), um die stärkeren Abweichungen entsprechend zu bestrafen. Zusätzlich werden deskriptive Verlustfunktionen vorgestellt, etwa die Anzahl der unterdrückten Käufe, die Anzahl der überflüssigen Artikel oder die Häufigkeit von fehlenden Beständen. Damit soll ermöglicht werden, die Qualität einer Bestellpolitik realistisch einschätzen zu können.

Im Einzelhandel sind die Kosten für das Unterschätzen deutlich höher als die Kosten für das Überschätzen, da man durch die Konkurrenzsituation stark auf das Wohlwollen der Kunden angewiesen ist. Es gibt aber auch Anwendungsfälle in denen die Relation umgekehrt ist (beispielsweise bei Ticketbestellungen für Fluggesellschaften).

3.2 Betriebswirtschaftliche Restriktionen

Bei der Beurteilung der Qualität einer Vorhersage ist nicht allein wichtig, dass der nächste Wert möglichst genau vorhergesagt wird. Es müssen auch bestimmte Eigenschaften des Anwendungsgebiets berücksichtigt werden, die die Benutzbarkeit einer Prognose beeinflussen.

Bei der Bestellung muss beispielsweise berücksichtigt werden, dass die Artikel eventuell nur in bestimmten Mengeneinheiten geliefert werden können. Während die Belieferung einer Filiale aus einem Zentrallager meist auf die Stückzahl genau ist, müssen bei der Bestellung von einem Lieferanten im Allgemeinen ganze Kartons oder Paletten abgenommen werden. Dies vereinfacht die Prognose, da man nur bis zu eine gewissen Genauigkeit erreichen muss und die Bestellmenge zur nächsten Bestelleinheit auf- oder abrunden kann. Dadurch fallen keine Kosten an, wenn der Fehler der Prognose unter der minimalen Bestellgröße liegt, insbesondere sind zwei Prognosen gleichwertig, wenn ihre maximale Abweichung vom tatsächlichen Wert dasselbe Vielfache der minimalen Bestelleinheit ist Ein ähnlicher Effekt ergibt sich daraus, dass im Geschäft selbst mehrere Einheiten eines Artikels im Regal oder in einem Handlager vorhanden sind, so dass geringe

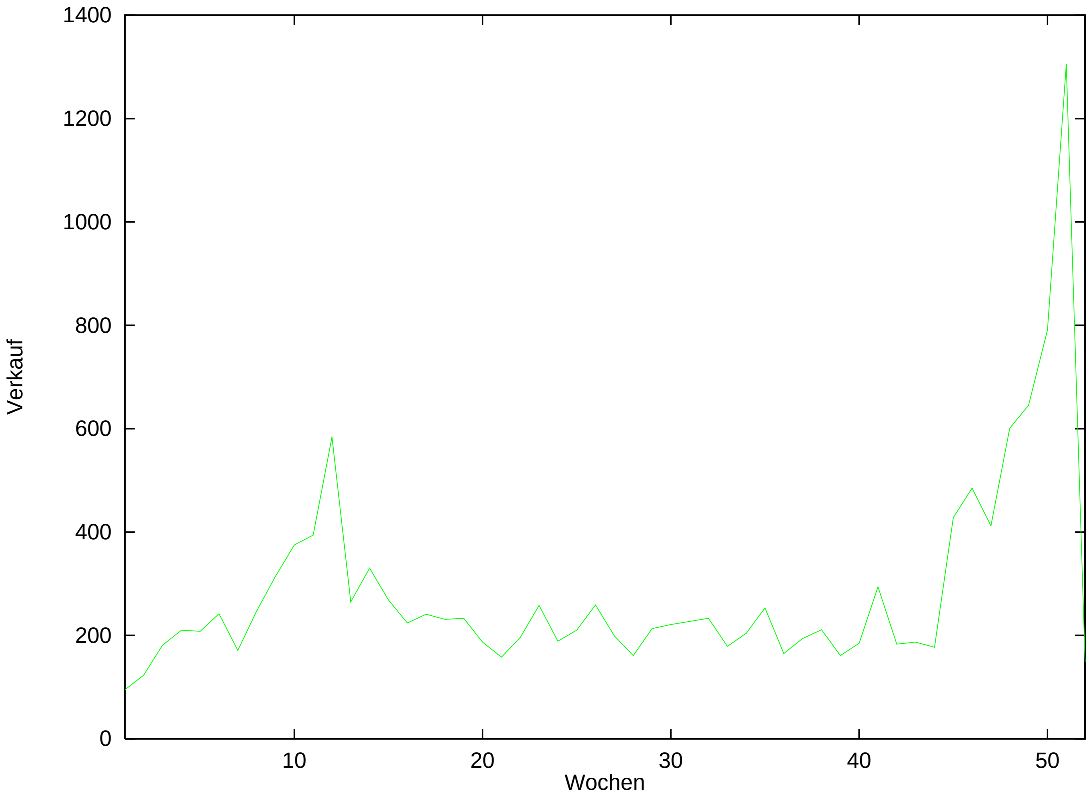


Abbildung 3.3: typischer Verlauf der Verkaufszahlen in einem Jahr

Schwankungen in der Versorgung durch das Zentrallager ausgeglichen werden können.

Eine andere Einschränkung besteht darin, dass die Artikel eine gewisse Lieferzeit benötigen. Abhängig von der jeweiligen Lieferzeit müssen die Daten für eine bestimmte Anzahl Wochen in der Zukunft prognostiziert werden. Ändert sich die Lieferzeit eines Artikels, so kann sich also auch die optimale Prognosemethode verändern.

3.3 Globale Parameter

Will man genauer Untersuchen welche Faktoren den Verkauf beeinflussen, so kann man unterscheiden ob sich diese Faktoren auf alle Artikel auswirken oder nur auf eine bestimmte Gruppe oder einzelne Artikel.

Zu den Faktoren, die sich auf alle Artikel auswirken, zählen zum Beispiel das Auftreten von Feiertagen und Ferien, das Wetter und die Größe und Lage der jeweiligen Filiale. Abbildung 3.3 zeigt zum Beispiel einen typischen Verlauf der Verkäufe eines Artikels in einem Jahr. Man sieht deutlich, wie sich die Verkaufszahlen zu Ostern und zu Weihnachten erheblich steigern.

Obwohl sich diese Faktoren global bemerkbar machen kann doch die Art ihrer Auswirkungen auf die einzelnen Artikel nicht global bestimmt werden. So ist es möglich, dass sich ein Artikel bei heißem Wetter sehr gut verkauft (z.B. Sonnencreme) während ein anderer Artikel bei heißem Wetter gar nicht verkauft wird (z.B. Hustenbonbons). Es muss also für jeden Artikel einzeln geprüft werden, wie sich die verschiedenen Parameter auf den Verkauf auswirken. Der Vorteil der globalen Parameter liegt also darin, dass ihre Existenz einfach aus einer Menge von Artikeln zu bestimmen ist, nicht darin, dass sich mit ihnen die Verkäufe unmittelbar prognostizieren lassen.

Einen besonders großen Einfluss auf die Verkaufszahlen hat die Weihnachtszeit. Dieser Einfluss kann sich auf verschiedene Weise manifestieren kann. Zum einen gibt es Artikel, die nur in der Weihnachtszeit verkauft werden, wie zum Beispiel Weihnachtsschmuck. Diese sind dadurch gekennzeichnet, dass der Bedarf einige Wochen vor Weihnachten einsetzt und sich bis zur Weihnachtswoche stetig steigert. Weiter gibt es typische Geschenkartikel, die das ganze Jahr über unregelmäßig verkauft werden, zu Weihnachten aber eine deutliche Spitze haben. Hierzu zählen beispielsweise Parfümerieartikel. Eine dritte Klasse bilden Artikel, die das restliche Jahr über gleichmäßig verkauft werden, kurz vor Weihnachten aber einen Anstieg erleben, da für die Feiertage Vorräte angelegt werden. Allen Gruppen gemeinsam ist der Effekt, dass nach Weihnachten deutlich weniger Artikel als normal eingekauft werden, da die Kunden ihre Weihnachtsvoräte erst aufbrauchen müssen. Für die Prognose müssen also nicht nur zukünftige sondern auch vergangene Feiertage berücksichtigt werden.

Ähnlich Effekte gibt es auch zu Ostern oder anderen Feiertagen sowie während der Ferienzeiten. Dabei hat nicht jeder Feiertag einen Einfluss auf jeden Artikel, manchmal sind es auch nur ganz spezielle Anlässe, zu denen die Artikel stark verkauft werden (z.B. Grabkerzen zu Allerheiligen).

Allgemein lassen sich fünf spezielle Saisonfiguren unterscheiden, nämlich Verkaufsspitzen im Frühjahr, zu Ostern, zu den Sommerferien, in der Weihnachtszeit, genau zu Weihnachten und der gleichmäßige Verkauf ohne spezielle Saison. Die genaue Ausprägung dieser Saisonfiguren ändert sich von Jahr zu Jahr.

Einen weiteren Einfluss hat das Wetter. Dieses wirkt sich zum Beispiel stark auf den Verkauf von Gartenmöbeln oder Hustenbonbons aus, deren Bedarf beim Einsetzen der entsprechenden Witterung stark steigt. Da allerdings zwischen der Bestellung und der Lieferung der Artikel eine oder mehrere Wochen vergehen, aber das Wetter über einen so langen Zeitraum nicht genau genug vorhergesagt werden kann, ist der Nutzen dieses Wissens zweifelhaft. Möglicherweise ist es günstiger, den Bedarf lediglich aufgrund des Datums vorherzusagen oder Wetterprognosen nur als zweitrangiges Prognosekriterium hinzu zunehmen.

Die Größe der Filiale, ihre Lage und die Struktur ihrer Kundschaft ist ebenfalls ein wichtiger Faktor für die Prognose der Verkaufszahlen, da sich das Einkommen und die soziale Situation der Kunden stark auf ihr Einkaufsverhalten auswirkt. Hier ist es sinnvoll, die Filialen in verschiedene Gruppen aufzuteilen und die Prognose für jede Gruppe oder auch jede Filiale einzeln zu stellen. Es ist aber auch denkbar, aus den Verkäufen verschiedener Filialen und Informationen über die Struktur der Filialen und ihrer Kundschaft etwas über das Einkaufsverhalten zu lernen und dieses Wissen mit in die Prognose einfließen zu lassen.

3.4 Artikelspezifische Parameter

Neben den globalen Einwirkungen gibt es auch Parameter, die sich nur auf einzelne Artikel auswirken. Hierzu zählen der Verkaufspreis des Artikels und das Auftreten von Werbemaßnahmen für den Artikel.

Es ist zu beachten, dass nicht nur das Auftreten sondern auch die Art der Werbemaßnahme berücksichtigt werden muss, da es hier viele unterschiedliche Formen gibt (spezielle Werbung für den Artikel in Prospekten / Zeitungen / Fernsehen; Werbung, die den Artikel unter mehreren anderen enthält; Werbung für das Geschäft im Allgemeinen; direkte Promotion in der Filiale; bessere Plazierung des Artikels).

Wichtig ist auch, dass Werbe- und Preissenkungsaktionen oft den Effekt haben, dass sich der Verkauf der Artikel nur verschiebt, d.h. nach Ende der Aktion fällt die Nachfrage unter den vorherigen Wert. Man muss hier also wiederum auch vergangene Aktionen berücksichtigen.

3.5 Artikelgruppen

Man sollte für die Prognose der Verkaufszahlen eines Artikels nicht nur den Artikel selbst sondern auch den Verkauf anderer Artikel berücksichtigen. Dies kann auf zwei Arten geschehen.

Zum einen gibt es Gruppen von Artikeln, die oft zusammen verkauft werden. Diese können mit verschiedenen Verfahren, beispielsweise dem Apriori-Algorithmus ([Agrawal et al., 1996]), identifiziert werden. Hier ist bei steigender Nachfrage nach einem Artikel auch mit steigender Nachfrage nach den anderen Artikeln dieser Gruppe zu rechnen. Werden die Artikel immer zusammen verkauft ist dies für die Prognose nicht unbedingt zu verwendbar, da ja auch die veränderte Nachfrage nach den anderen Artikeln der Gruppe erkannt werden muss, möglicherweise sind die Verkäufe der einen Artikel aber aus irgendwelchen Gründen besser zu prognostizieren als die Verkäufe der anderen, so dass sich hier eine Verbesserung der Vorhersage eines Teils der Artikel ergibt.

Andererseits gibt können sich die Artikel in einer Warengruppe auch derart beeinflussen, dass der erhöhte Verkauf eines Artikels den verminderten Verkauf eines anderen Artikel nach sich zieht. Dies ist der Fall, wenn mehrere Artikel äquivalent zueinander sind, beispielsweise verschiedene Arten von Spülmittel oder ein Artikel in mehreren Verpackungsgrößen. Wenn man davon ausgehen kann, dass der Gesamtbedarf an diesen Artikeln konstant ist oder man den Gesamtbedarf unabhängig vom Verkauf der einzelnen Artikel vorhersagen kann, so kann man die einzelnen Prognosen miteinander vergleichen und aufeinander abstimmen. Dies kann zum Beispiel geschehen, indem man die artikelspezifischen Parameter anderer Artikel mit in die Vorhersage aufnimmt.

Ein Spezialfall davon ist, wenn ein neuer Artikel eingeführt wird, der ältere Artikel ersetzt oder ergänzen soll. Da für diesen Artikel keine Daten vorliegen kann versucht werden, diesen aufgrund Verkäufe in der Artikelgruppe vorherzusagen. Man nimmt dann an, dass sich allgemeine Effekte auf alle Artikel dieser Gruppe gleich auswirken.

Gelegentlich gibt es auch Artikel, deren Verkäufe die Verkäufe anderer Artikel zu einem späteren Zeitpunkt beeinflussen. Hat man solche Vorläuferartikel einmal gefunden, so kann man die spätere Prognose bereits aufgrund gesicherter Daten erstellen.

3.6 Kurze Zeitreihen

Im Einzelhandel tritt speziell das Problem auf, dass die vorhandenen Zeitreihen der Verkäufe kurz sind. Dies hat zwei Gründe: Zum einen werden pro Jahr ca. 20% neue Artikel eingeführt während andere Artikel verschwinden, zum anderen ist das Käuferverhalten sehr dynamisch, so dass ältere Daten nur eingeschränkt zur Vorhersage neuerer Verkäufe vorhergesagt werden können.

Die neu eingeführten Artikel sind allerdings nur zum Teil wirklich neue Arten von Artikeln, oft wird auch nur ein zu bekannten Artikeln ähnlicher Artikel eingeführt oder ein alter Artikel durch einen leicht verbesserten ersetzt. In vielen Fällen ist es also möglich die Vorhersage neuer Artikel auf bekannte Daten zu stützen. Allerdings sollte die Prognosemethode so gewählt werden, dass neuere Daten einen größeren Einfluss auf die Prognose haben, so dass Veränderungen in den Verkäufen schnell erkannt werden können. Dies wird allerdings dadurch erschwert, dass die Neueinführung eines Artikels oft von Werbemaßnahmen begleitet wird, so dass grundlegende Änderungen im Verkaufsmuster des Artikels zunächst schlecht erkannt werden können.

Ähnliches gilt auch für die Benutzung älterer Verkaufsdaten in der Prognose. Da sich das Käuferverhalten ständig ändert kann eine Vorhersage die zu alte Daten einbezieht zu einer Fehlprognose führen. Daher ist es günstig sehr alte Daten wegzulassen oder die Prognose so einzurichten, dass sie sich in erster Linie nach den neueren Daten richtet.

Allerdings muss man hier zwischen zwei entgegengesetzten Einflüssen abwägen. Während alte Daten bei einer Veränderung des Kaufverhaltens die Prognose negativ beeinflussen sind sie doch nützlich um zufällige Schwankungen in den Verkäufen von signifikanten wiederkehrenden Effekten zu unterscheiden. Falls kein Möglichkeit existiert, Veränderungen im Kaufverhalten direkt ausfindig zu machen, sind zwei Auswege aus diesem Problem denkbar. Der triviale Weg wäre, empirisch herauszufinden bis zu welchem Zeitraum und in welchem Maß man alte Daten zur Prognose berücksichtigen sollte. Da man dabei aber erwarten kann, dass sich für jeden Artikel andere Größen ergeben, ist dieser Weg für die praktische Umsetzung schlecht geeignet.

Die erfolgversprechendere Möglichkeit zwischen zufälligen und signifikanten Schwankungen zu unterscheiden ist die, zu versuchen die signifikanten Prozesse aus den Verkäufen mehrerer Artikel herauszusuchen. Dazu kann man die Zeitreihen clustern und erhält durch die Beschreibung der Cluster mehrere prototypische Zeitreihen entdecken. Änderungen im Käuferverhalten sollten sich in allen Zeitreihen eines Clusters bemerkbar machen, während zufällige Schwankungen auf eine Zeitreihe beschränkt sind. Benutzt man die prototypischen Zeitreihen aus den Clustern als Daten für die Vorhersage eines Artikels kann man herausfinden,wie sehr dieser Artikel jeweils den anderen Artikel gleicht.

Kapitel 4

Asymmetrische Kostenfunktionen in der SVM

Außer der linearen und der quadratischen ε-insensitiven Verlustfunktion sind auch andere Kostenfunktionen möglich, wobei im allgemeinen nur die Konvexität der Verlustfunktion vorausgesetzt werden muss. In [Smola et al., 1998] werden mögliche andere symmetrische Verlustfunktionen betrachtet.

In der Praxis treten aber oft Situationen auf, in denen die Kosten für eine Fehlprognose nicht symmetrisch sind. Dafür kann die Support Vector Machine um asymmetrische Kostenfunktionen erweitert werden, indem man jeweils für die positive und die negative Abweichung einen linearen oder quadratischen Verlust mit verschiedenen Vorfaktoren benutzt.

Daraus ergeben sich drei Typen von Verlustfunktionen (Abbildung 4.1), die Lin-lin-Verlustfunktion (beide Äste linear), die Quad-quad-Verlustfunktion (beide Äste quadratisch) und die Lin-quad-Verlustfunktion (ein Ast linear und einer quadratisch). Jede dieser Verlustfunktionen ist jeweils noch durch die Vorfaktoren L_{pos} und L_{neg} gekennzeichnet.

Zusätzlich kann man die Verlustfunktionen noch ε-invariant gestalten mit verschiedenen ε_+ und ε_- für positive und negative Abweichungen. Hier genügt es aber den Fall $\varepsilon_+ = \varepsilon_-$ zu betrachten, da man ansonsten die Daten durch $y_i \mapsto y_i + \frac{1}{2}(\varepsilon_+ - \varepsilon_-)$ transformieren kann und von den vorhergesagten Werten wieder $\frac{1}{2}(\varepsilon_+ - \varepsilon_-)$ abzieht.

4.1 Lin-lin-Verlustfunktion

Die $(L_{pos}, L_{neg}, \varepsilon)$-Lin-lin-Verlustfunktion ist definiert durch

$$L(y, f(x,\alpha)) = \begin{cases} 0 & \text{falls } |y - f(x,\alpha)| \leq \varepsilon \\ L_{neg} \cdot (y - f(x,\alpha) - \varepsilon) & \text{falls } f(x,\alpha) < y - \varepsilon \\ L_{pos} \cdot (f(x,\alpha) - y - \varepsilon) & \text{falls } f(x,\alpha) > y + \varepsilon \end{cases}$$

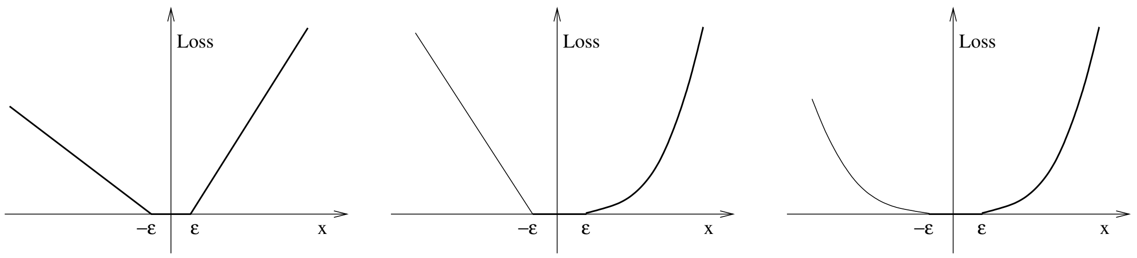


Abbildung 4.1: Lin-lin-, lin-quad- und quad-quad- Verlustfunktion.

Die Zielfunktion ist hier für einen gegebenen Wert $\hat{C}$ gegeben durch

$$\Phi(w,\xi,\xi^*) = \frac{1}{2}(w^T w) + C^* \sum_{i=1}^{l} \xi^* + C \sum_{i=1}^{l} \xi$$

Dabei sei $C = L_{pos} \cdot \hat{C}$ und $C^* = L_{neg} \cdot \hat{C}$.

Φ soll wieder unter den Nebenbedingungen

$$\begin{aligned}
y_i - (w^T x_i) - b &\leq \varepsilon + \xi_i^* \quad , i = 1,\dots,n \quad \text{(negative Abweichung}^1\text{)} && (4.1)\\
(w^T x_i) + b - y_i &\leq \varepsilon + \xi_i \quad , i = 1,\dots,n \quad \text{(positive Abweichung)} && (4.2)\\
\xi_i^* &\geq 0 \quad , i = 1,\dots,n && (4.3)\\
\xi_i^* &\geq 0 \quad , i = 1,\dots,n && (4.4)
\end{aligned}$$

minimiert werden. Analog zum einfach linearen Fall ergibt sich hier, dass die Minimierung von Φ äquivalent ist zur Maximierung von

$$\begin{aligned}
W(\alpha,\alpha^*) &= -\frac{1}{2}\sum_{i,j=1}^{l} (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)(x_i \cdot x_j)\\
&\quad + \sum_{i=1}^{n} y_i(\alpha_i^* - \alpha_i) - \sum_{i=1}^{n} \varepsilon(\alpha_i^* + \alpha_i) && (4.5)
\end{aligned}$$

unter den Nebenbedingungen

$$\begin{aligned}
0 \leq \alpha_i^* \leq C^* & \quad , i = 1,\dots,n && (4.6)\\
0 \leq \alpha_i \leq C & \quad , i = 1,\dots,n && (4.7)\\
\sum_{i=1}^{n} \alpha_i^* &= \sum_{i=1}^{n} \alpha_i && (4.8)
\end{aligned}$$

Der Unterschied besteht also lediglich darin, dass für die Variablen α_i und α_i^* unterschiedliche obere Schranken gelten.

4.2 Quad-quad-Verlustfunktion

Die $(L_{pos}, L_{neg}, \varepsilon)$-Quad-quad-Verlustfunktion ist definiert durch

$$L(y, f(x,\alpha)) = \begin{cases} 0 & \text{falls } |y - f(x,\alpha)| \leq \varepsilon \\ L_{neg}(y - f(x,\alpha) - \varepsilon)^2 & \text{falls } f(x,\alpha) < y - \varepsilon \\ L_{pos}(f(x,\alpha) - y - \varepsilon)^2 & \text{falls } f(x,\alpha) > y + \varepsilon \end{cases}$$

Die Zielfunktion ist gegeben durch

$$\Phi(w,\xi,\xi^*) = \frac{1}{2}(w^T w) + \frac{1}{2} C^* \sum_{i=1}^{l} \xi^{*2} + \frac{1}{2} C \sum_{i=1}^{l} \xi^2$$

und ist unter den Nebenbedingungen 4.1 - 4.4 zu minimieren. Analog zum symmetrischen quadratischen Fall ergibt sich

$$\begin{aligned}
L(w,b,\xi,\xi^*;\alpha,\alpha^*,\gamma,\gamma^*) &= W(\alpha,\alpha^*) && (4.9)\\
&= -\frac{1}{2}\sum_{i,j=1}^{l} (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)(x_i \cdot x_j)\\
&\quad + \sum_{i=1}^{n} y_i(\alpha_i^* - \alpha_i) - \sum_{i=1}^{n} \varepsilon(\alpha_i^* + \alpha_i)\\
&\quad - \frac{1}{C^*}\sum_{i=1}^{n} \alpha_i^{*2} - \frac{1}{C}\sum_{i=1}^{n} \alpha_i^2 && (4.10)
\end{aligned}$$

Es ist also $W(\alpha, \alpha^*)$ zu maximieren unter den Nebenbedingungen

$$0 \leq \alpha_i^*, \alpha_i \qquad , i = 1, \ldots, n \tag{4.11}$$

$$\sum_{i=1}^{n} \alpha_i^* = \sum_{i=1}^{n} \alpha_i \tag{4.12}$$

Der Unterschied zum symmetrischen Fall besteht hier in unterschiedlichen Konstanten C und C^* in den Faktoren von α und α^* im quadratischen Term der Zielfunktion.

4.3 Lin-quad-Verlustfunktion

Die $(L_{pos}, L_{neg}, \varepsilon)$-Lin-quad-Verlustfunktion ist definiert durch

$$L(y, f(x,\alpha)) = \begin{cases} 0 & \text{falls } |y - f(x,\alpha)| \leq \varepsilon \\ L_{neg}(y - f(x,\alpha) - \varepsilon) & \text{falls } f(x,\alpha) < y - \varepsilon \\ L_{pos}(f(x,\alpha) - y - \varepsilon)^2 & \text{falls } f(x,\alpha) > y + \varepsilon \end{cases}$$

Dieser Fall ist eine Kombination von linearen und quadratischen Kostenfunktionen. Die Zielfunktion ist gegeben durch

$$\Phi(w, \xi, \xi^*) = \frac{1}{2}(w^T w) + \frac{1}{2}C^* \sum_{i=1}^{l} \xi^{*2} + C \sum_{i=1}^{l} \xi$$

und ist unter den Nebenbedingungen 4.1 - 4.4 zu minimieren. Analog zum linearen (für ξ) bzw. quadratischen Fall (für ξ^*) erhält man, dass man um Φ zu minimieren die Funktion $W(\alpha^*, \alpha)$ maximieren muss

$$\begin{aligned} W(\alpha, \alpha^*) &= -\frac{1}{2} \sum_{i,j=1}^{l} (\alpha_i^* - \alpha_i)(\alpha_j^* - \alpha_j)(x_i \cdot x_j) \\ &\quad + \sum_{i=1}^{n} y_i(\alpha_i^* - \alpha_i) - \sum_{i=1}^{n} \varepsilon(\alpha_i^* + \alpha_i) \\ &\quad - \frac{1}{2C^*} \sum_{i=1}^{n} \alpha_i^{*2} \end{aligned} \tag{4.13}$$

unter den Nebenbedingungen

$$0 \leq \alpha_i^* \qquad , i = 1, \ldots, n \tag{4.14}$$

$$0 \leq \alpha_i \leq C \qquad , i = 1, \ldots, n \tag{4.15}$$

$$\sum_{i=1}^{n} \alpha_i^* = \sum_{i=1}^{n} \alpha_i \tag{4.16}$$

Kapitel 5

Anwendung

5.1 Der betrachtete Anwendungsfall

5.1.1 Die Daten

Die hier betrachteten Daten sind Verkaufszahlen von Artikeln einer Einzelhandelskette. Gegeben sind die wöchentlichen Verkäufe von 24486 Artikeln aus 20 Filialen in der Zeit von der 48. Woche 1995 bis zur 52. Woche 1997. Zusätzlich sind pro Artikel das Ein- und Auslistungsdatum bekannt sowie die Einteilung in zwei Warengruppen und in einigen Fällen der Verkaufspreis des Artikels.

Es wurden nur die Artikel betrachtet, die während des gesamten Zeitraum im Verkauf waren. Artikel mit falschen Verkaufszahlen (in einigen Fällen war z.B. der Eintrag „99999“ vorhanden) wurden entfernt.

Aus den Artikeln wurden zwei Gruppen herausgesucht. Die erste Gruppe bilden die 50 Artikel, die im Durchschnitt am häufigsten verkauft wurden. Die durchschnittliche Anzahl der Verkäufe pro Woche lag hier zwischen 33.69 und 366.66. Aus allen 20 Filialen ergaben sich nach Entfernen der fehlerhaften Daten 906 Zeitreihen. Abbildung 5.1 zeigt die durchschnittliche wöchentlichen Verkäufe in dieser Artikelgruppe.

Die zweite Gruppe bilden 50 zufällig ausgewählte Artikel mit geringer durchschnittlicher Verkaufszahl, diese liegen hier zwischen 3.58 und 3.71. Insgesamt ergaben sich hier 859 Zeitreihen. Die durchschnittlichen wöchentlichen Verkäufe der Artikel in dieser Gruppe sind in Abbildung 5.2 abgebildet.

Zum Vergleich sind beide Verkaufskurven noch einmal in Abbildung 5.3 zusammengefasst, man sieht, dass die Verkäufe in Gruppe I deutlich größer sind als die in der Gruppe II. Insbesondere sind zwei Dinge bemerkenswert: Zum einen erkennt man in beiden Zeitreihen deutlich den Anstieg der Verkäufe zu Weihnachten, der etwa Anfang Dezember einsetzt und in der Weihnachtswoche (Wochen 4, 56 und 108) seinen Höhepunkt erreicht. Direkt nach Weihnachten ist ein starker Abfall der Verkäufe unter das Niveau vor Weihnachten zu erkennen. Zum anderen ist in den Zeitreihen ein positiver Trend zu erkennen (Abbildung 5.4).

Leider sind die deutlichen Saisonfiguren, die in den durchschnittlichen Verkäufen zu sehen sind, in den Zeitreihen der einzelnen Artikel oft nicht zu erkennen. Die Abbildungen 5.5 und 5.6 zeigen typische Zeitreihen in den Gruppen I und II. Hier kann die Zeitreihe wahrscheinlich nur durch durch ihren Erwartungswert und ihre Varianz charakterisiert werden, so dass eine relativ einfache Prognosemethode ausreicht.

5.1.2 Die Verlustfunktion

Das Problem der optimalen Prognose ist durch die betrachteten Daten und die zu verwendende Verlustfunktion gegeben. Da die spätere Optimierung unmittelbar auf der Verlustfunktion aufbaut sollte man hier besonders sorgfältig eine Funktion wählen, die den betriebswirtschaftlichen Gegebenheiten so genau wie möglich entspricht. Die jeweilige Kostenfunktion ist einzig aus der

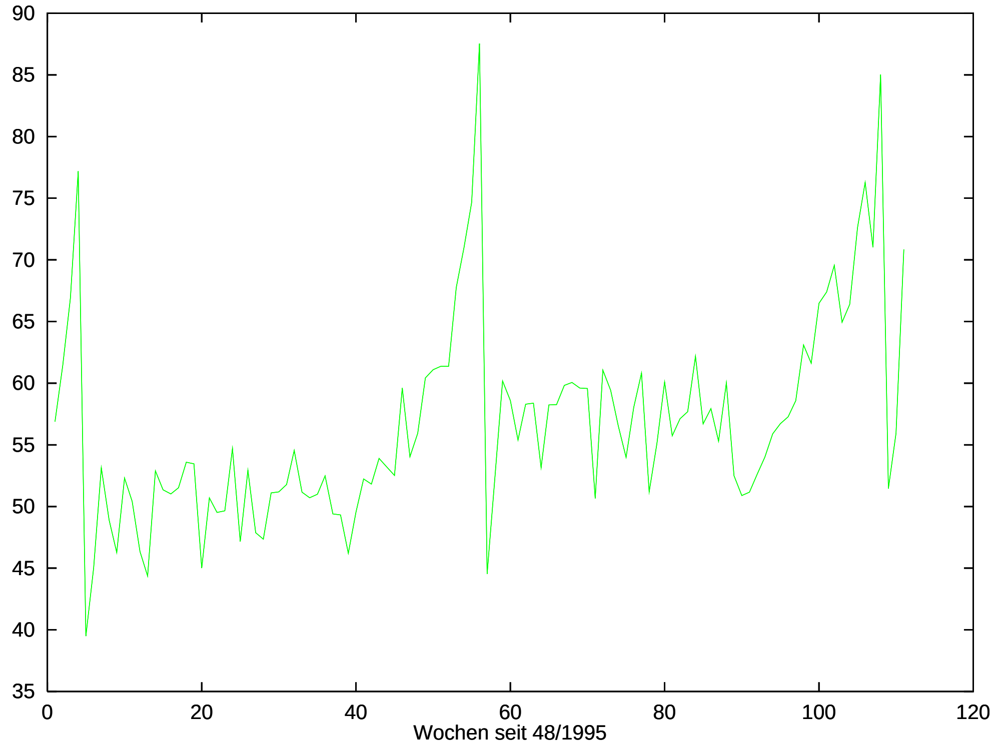


Abbildung 5.1: Durchschnittliche wöchentliche Verkäufe der Artikelgruppe I

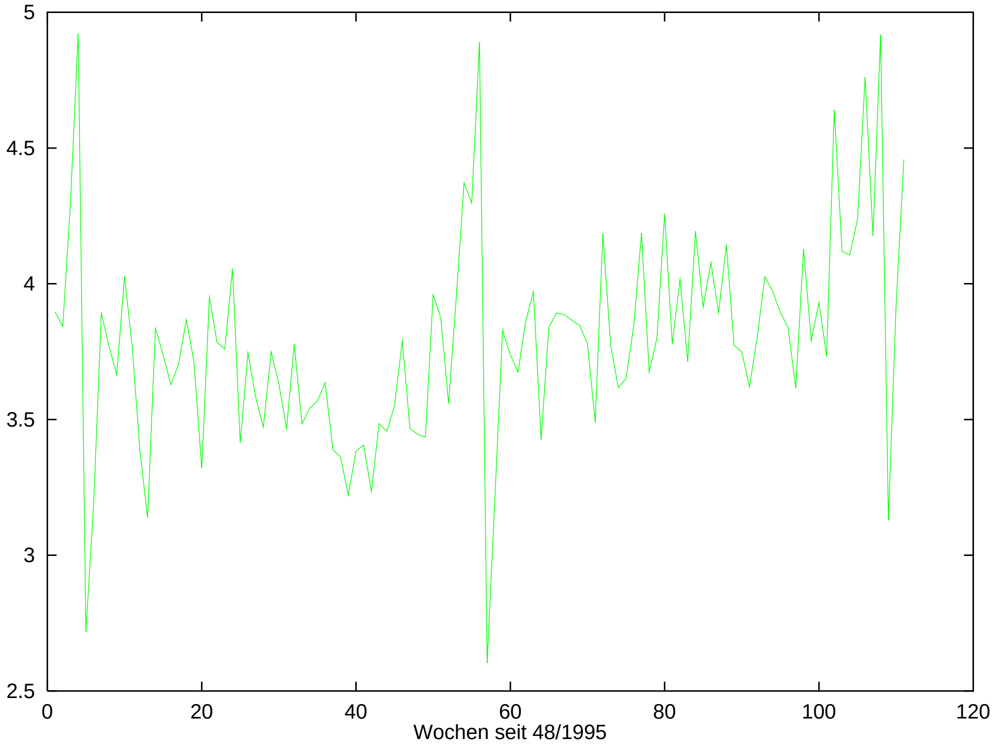


Abbildung 5.2: Durchschnittliche wöchentliche Verkäufe der Artikelgruppe II

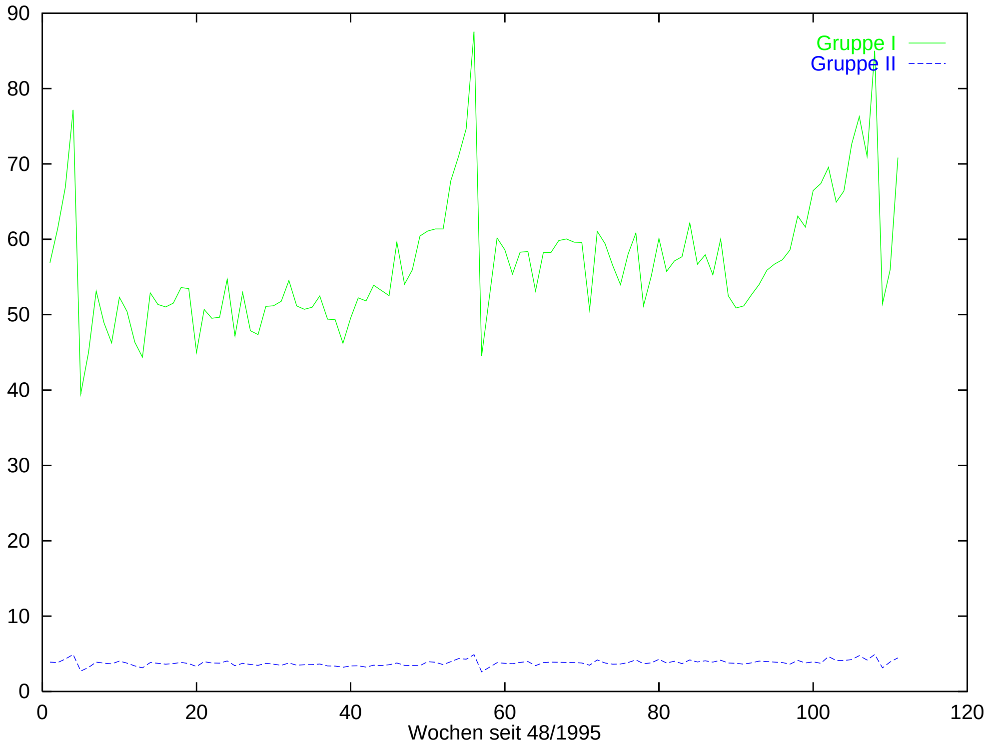


Abbildung 5.3: Vergleich der durchschnittlichen wöchentlichen Verkäufe der Artikelgruppen I-II

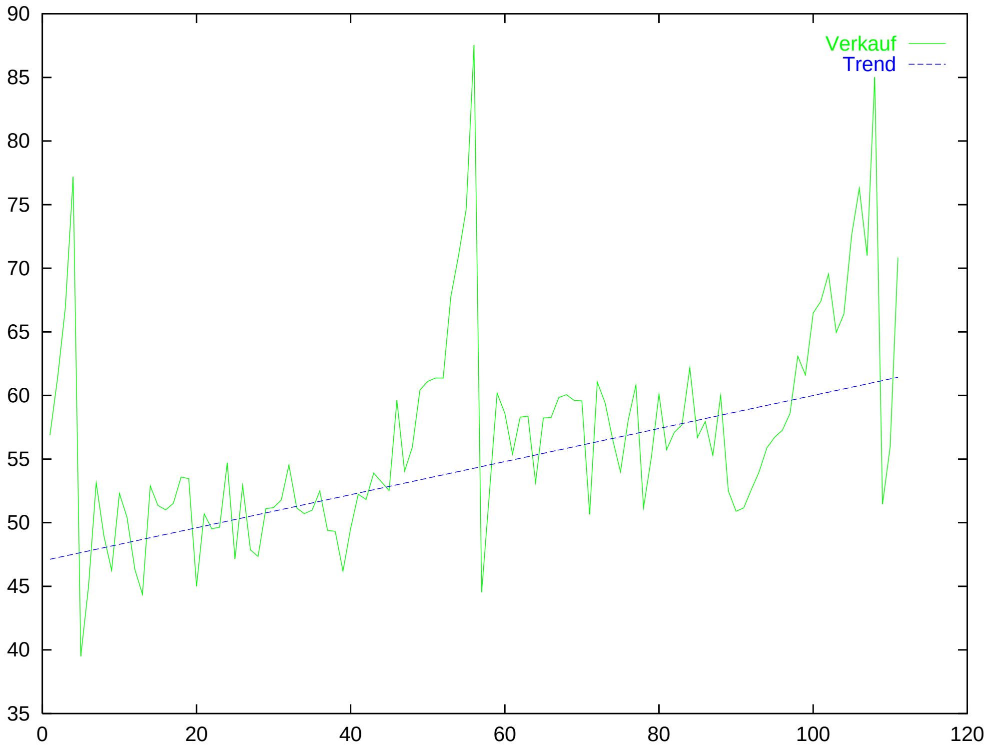


Abbildung 5.4: Verkäufe der Artikelgruppe I und Trend

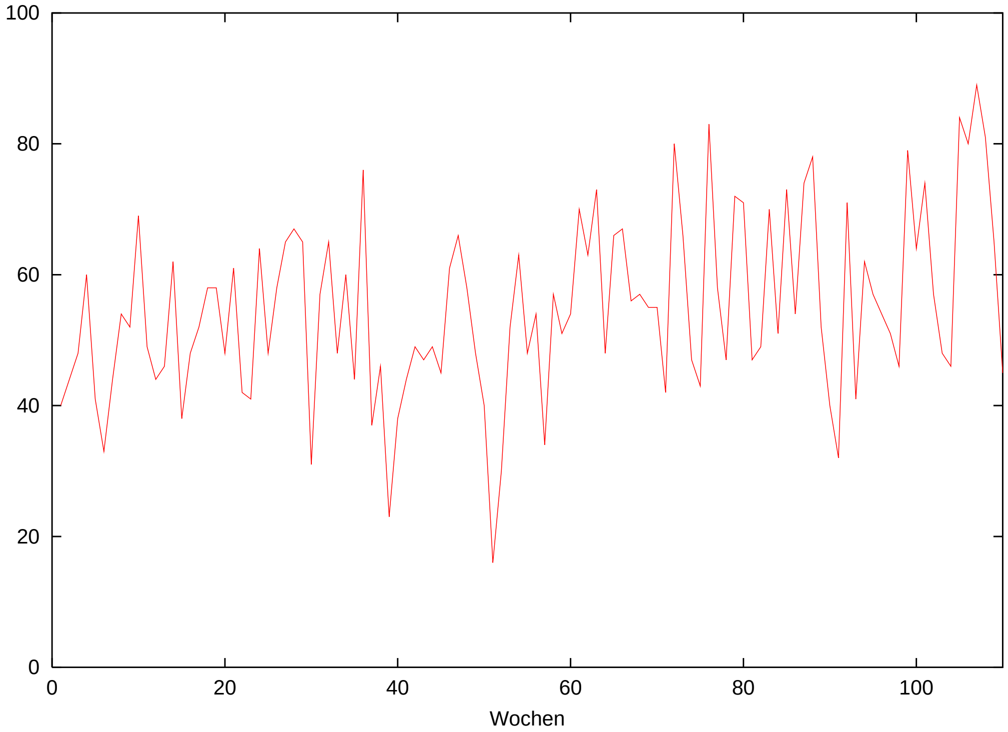


Abbildung 5.5: Typische Zeitreihe in der Artikelgruppen I

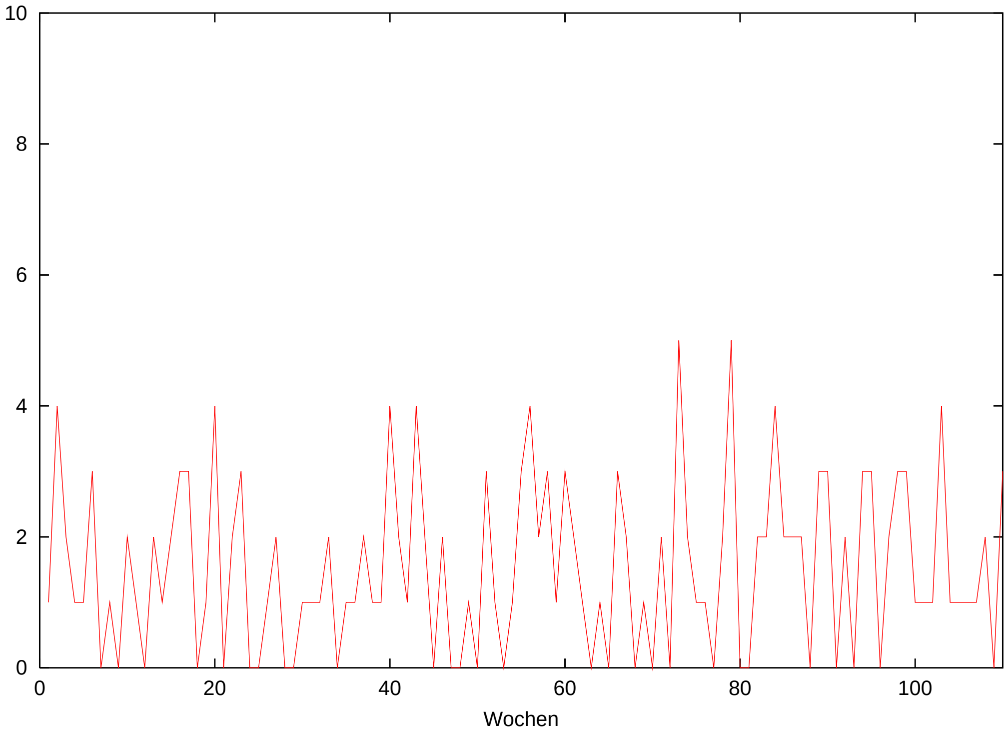


Abbildung 5.6: Typische Zeitreihe in der Artikelgruppen II

Anwendung gegeben und setzt eine gute Kenntnis der Prozesse des Unternehmens voraus, was in der Realität nicht immer gegeben ist.

Bemerkenswert ist noch, dass die Multiplikation der Kostenfunktion mit einer positiven Zahl das Ergebnis nicht verändert, sondern lediglich dafür sorgt, daß der minimal zu erreichende Durchschnitts- oder Gesamtverlust ebenfalls mit diesem Faktor skaliert wird. Misst man den Verlust also in entgangenem Geld, so ist die dabei verwendete Währung nicht von Bedeutung.

Wie in Abschnitt 3.1 erläutert sind als Kostenfunktionen asymmetrische lineare oder quadratische Funktionen geeignet, wobei die jeweiligen Faktoren durch die anfallenden Kosten bei einer Fehlprognose gegeben sind. Nach Abschnitt 3.2 kann es sinnvoll sein, für eine kleine Abweichung der Prognose keine Kosten anzusetzen. Im weiteren wird eine lineare Kostenfunktion der folgenden Form verwendet:

$$L(y_t, \hat{y}_t) = \begin{cases} 0 & \text{falls } |y - \hat{y}_t| \leq \varepsilon \\ L_{neg} \cdot (y_t - \hat{y}_t - \varepsilon) & \text{falls } \hat{y}_t < y_t - \varepsilon \\ L_{pos} \cdot (\hat{y}_t - y_t - \varepsilon) & \text{falls } \hat{y}_t > y_t + \varepsilon \end{cases}$$

Dabei bezeichnet $\hat{y}_t$ die Vorhersage für den Zeitpunkt t und y_t den tatsächlich eingetretenen Wert.

Laut [Arminger und Götz, 1999] kann man den Faktor L_{neg} für den Verlust durch Unterschätzen des Bedarf für einen Artikel i durch die durchschnittliche Profitspanne m des Unternehmens und den Verkaufspreis s_i des Artikels als $L_{neg} = s_i \cdot m$ ansetzen. Der Faktor L_{pos} für das Überschätzen des Bedarfs ergibt sich aus einem betriebsinternen Zinssatz r als $L_{pos} = s_i \cdot r/52$ [1].

Mögliche Werte sind $m = 0.05, m = 0.10, m = 0.15$ und $m = 0.20$ sowie $r = 0.10, r = 0.20$ und $r = 0.30$. Weiter wurde $\varepsilon = 0.5$ gewählt, d.h. die Voraussage sollte bis auf einen Artikel gerundet genau sein.

Wie bereits erwähnt ändert die Multiplikation mit einer positiven Zahl das Ergebnis des Problems nicht. Daher ist in diesem Fall lediglich das Verhältnis L_{pos}/L_{neg} wichtig, insbesondere ist, wenn nur ein Artikel betrachtet wird, der Verkaufspreis s_i unbedeutend. Aus den angegebenen Werten ergibt sich $0.009 \leq L_{pos}/L_{neg} \leq 0.115$. Hier wurde $L_{pos}/L_{neg} = 0.05$ gewählt, d.h. die endgültige Verlustfunktion hat folgende Gestalt:

$$L(y_t, \hat{y}_t) = \begin{cases} 0 & \text{falls } |y - \hat{y}_t| \leq 0.5 \\ 20 \cdot (y_t - \hat{y}_t - \varepsilon) & \text{falls } \hat{y}_t < y_t - 0.5 \\ \hat{y}_t - y_t - \varepsilon) & \text{falls } \hat{y}_t > y_t + 0.5 \end{cases}$$

5.2 Statistische Standardverfahren

Zuerst soll versucht werden, die Zeitreihen mit einem bekannten Verfahren zu prognostizieren. Die Wahl fiel hier auf das Verfahren der exponentiellen Glättung. Dieses Verfahren schätzt die Werte einer Zeitreihe $(X_1, X_2, X_3, \ldots)$ mittels eines Glättungsparameter λ durch:

$$\hat{X}_{t+1} = \lambda * X_t + (1 - \lambda) * \hat{X}_t$$

Dieses Verfahren ist wegen seinem geringen Speicherplatzbedarf und der formalen Einfachheit im ökonomischen Bereich weit verbreitet (siehe [Schlittgen und Streitberg, 1987], Abschnitt 7.3.4).

Um die Asymmetrie der Aufgabenstellung zu berücksichtigen wurde die Prognose dann mit einer Konstanten C multipliziert. Die Motivation dazu liefert die in der Praxis oft benutzte betriebswirtschaftliche Methode immer doppelt so viele Artikel wie benötigt im Lager zu halten, um jederzeit über eine Sicherheitsreserve zu verfügen.

Für jede der beiden Artikelgruppen wurde jeweils ein Versuch durchgeführt, in dem der Glättungparameter λ auf dem ersten Jahr optimal bestimmt wurde und ein Versuch, in dem sowohl der Glättungparameter λ als auch die Konstante C auf dem ersten Jahr optimal bestimmt wurde. Die optimalen Parameter wurden jeweils mit einer Genauigkeit von ± 0.01 bestimmt. Dann wurde

[1] Der Faktor 1/52 ergibt sich, da die Daten pro Woche vorliegen, d.h es gibt 52 Verkaufsperioden pro Jahr.

auf dem zweiten Jahr mit diesen Parametern der durchschnittliche Verlust bestimmt.

Verfahren	Parameter	Gruppe	Verlust
exp. Glättung	$\lambda_{opt} = 0.36 \quad C = 2$	I	66.447
exp. Glättung	$\lambda_{opt} = 0.37 \quad C_{opt} = 1.54$	I	52.407
exp. Glättung	$\lambda_{opt} = 0.09 \quad C = 2$	II	5.758
exp. Glättung	$\lambda_{opt} = 0.08 \quad C_{opt} = 2.05$	II	5.742

Es fällt auf, dass für die Artikelgruppe II der Parameter λ sehr klein gewählt werden muss, d.h. die Verkäufe in dieser Gruppe hängen sehr wenig vom letzten Wert ab. Im Vergleich dazu ist in der ersten Artikelgruppe λ hoch, es ist jeweils mit großen Änderungen der Verkäufe zu rechnen. Dies ist dadurch zu erklären, dass oft verkaufte Artikel stark an saisonalen und andere Einflüsse gekoppelt sind, während zufällige Schwankungen sich im Mittel über viele Kunden gegenseitig aufheben. Dadurch muss die Prognose die Änderungen in den Verkäufen stark berücksichtigen, um einen steigenden oder sinkenden Bedarf möglichst schnell nachzuvollziehen.

Bei selten verkauften Artikeln ist eher damit zu rechnen, dass hauptsächlich zufällig Einflüsse bestimmen, wie oft der Artikel in einer Woche verkauft wird. Daher sind die letzten Verkaufszahlen für eine Prognose weniger aussagekräftig als ein langfristiger Durchschnitt.

Eine weitere Untersuchung zeigte, dass der Versuch, die optimalen Parameter C und λ nach einer gewissen Zahl von betrachteten Beispielen neu zu bestimmen, keine wesentliche Verbesserung in der Vorhersage brachte. Bestimmt man die Parameter jeweils nach vier vorhergesagten Beispielen neu, so ändern weichen die Parameter C und λ jeweils um maximal 0.02 von den oben bestimmten Werten ab.

5.3 Durchführung der Versuche

Die Versuche wurden im Allgemeinen wie folgt durchgeführt: Für jeden Artikel wurden die letzten 52 Wochen (das Jahr 1997) in Abschnitte von vier Wochen aufgeteilt. Die Verkäufe dieser Wochen wurden jeweils vorhergesagt, indem die SVM mit den jeweiligen Attributen auf den vorhergehenden 52 Wochen trainiert wurde. Als Ergebnis wurde der durchschnittliche Verlust auf den getesteten Daten zurückgegeben.

Da die Versuche zeigten, dass die Kapazitätskonstante C der Support Vector Machine auf einem relativ großen Bereich - etwa 10^{-4} bis 10^{2} - keine großen Änderungen des Verlustes nach sich zogen, wurden alle Versuche mit der festen Konstante $C = 10^{-2}$ durchgeführt.

Um die numerische Stabilität des Optimierungs-Algorithmus zu garantieren wurden alle Attribute individuell in das Intervall $[-1, 1]$ skaliert. Dies verhindert, dass Attribute mit einem großen Wertebereich gegenüber Attributen mit kleineren Wertebereichen überbewertet werden.

Die Experimente wurden mit einer modifizierten Version der SVM–Software des Computer Learning Research Center der Royal Holloway, University of London durchgeführt ([Saunders et al., 1998]). Diese Software wurde um die Behandlung asymmetrischer Verlustfunktionen ergänzt.

Kapitel 6

Prognose in Warenwirtschaftssystemen

Die Support Vector Machine wurde bereits in [Mukherjee et al., 1997] und [Müller et al., 1997] erfolgreich zur Prognose von Zeitreihen mit symmetrischen Kostenfunktionen eingesetzt. In diesem Kapitel soll untersucht werden, wie die Support Vector Machine zur Prognose von Zeitreihen eingesetzt werden kann, um die in Kapitel 3 beschriebenen Problem zu lösen. Dabei soll nicht nur untersucht werden, wie die Qualität der Prognose zu verbessern ist. Es soll auch berücksichtigt werden, dass in der Praxis nur eine begrenzte Rechenzeit und Computerkapazität zur Verfügung steht. Da für das komplette Lernen aller 25000 Artikel bei einer Lernzeit von ca. einer Sekunde pro Artikel etwa sieben Stunden pro Filiale dauern würde, fällt die Möglichkeit, für jede neue Prognose einen neuen Lernlauf zu starten, aus.

6.1 Prognose von Zeitreihen mit der Support Vector Machine

Für die Prognose einer Zeitreihe $X_1, X_2, X_3, \cdots$ ist es notwendig, die Trainingsdaten $X_1, \cdots, X_K$ in Trainingsbeispiele für die Support Vector Machine umzuformen. Da die Support Vector Machine nur Beispiele mit einer festen Anzahl von Attributen bearbeiten kann, wählt man eine Konstante $n < K$ und benutzt die Trainingsbeispiele $(X_1, \ldots, X_n | X_{n+1}), \ldots, (X_{K-n}, \ldots, X_{K-1} | X_K)$[1]. Eventuell hat man auch einen Prognosehorizont h zu berücksichtigen, d.h. man benutzt die Beispiele $(X_1, \ldots, X_n | X_{n+h}), \ldots, (X_{K-h-n+1}, \ldots, X_{K-h} | X_K)$. Abbildung 6.1 illustriert den Aufbau eines Trainingsbeispiels.

Diese Attribute erlauben es, aus dem vergangenen Verhalten der Zeitreihe die zukünftige Entwicklung zu schätzen. Diese Attribute reichen aber in vielen Fällen nicht aus, um den Verlauf der Zeitreihe genau genug zu bestimmen. Dies kann daran liegen, dass die Zeitreihe anderen Einflüssen unterliegt, die nicht direkt von den letzten Verkäufen abhängen, oder einfach daran, dass die Grösse n des „Gedächtnisses" der Zeitreihe nicht groß genug ist.

Im klassischen Komponentenmodell der Zeitreihenanalyse (siehe [Schlittgen und Streitberg, 1987], Kapitel 1.3) wird unterstellt, dass die Zeitreihe aus vier Komponenten besteht

1. dem Trend, einer langfristigen systematischen Veränderung des mittleren Niveaus der Zeitreihe,

2. einer Konjunkturkomponente, die eine mehrjährige, nicht notwendig regelmäßige Schwankung darstellt,

[1] Die Schreibweise $(\bar{x}|y)$ besagt, dass mit den Attributen $\bar{x} \in \mathbf{R^n}$ der Wert $y \in \mathbf{R}$ vorhergesagt werden soll.

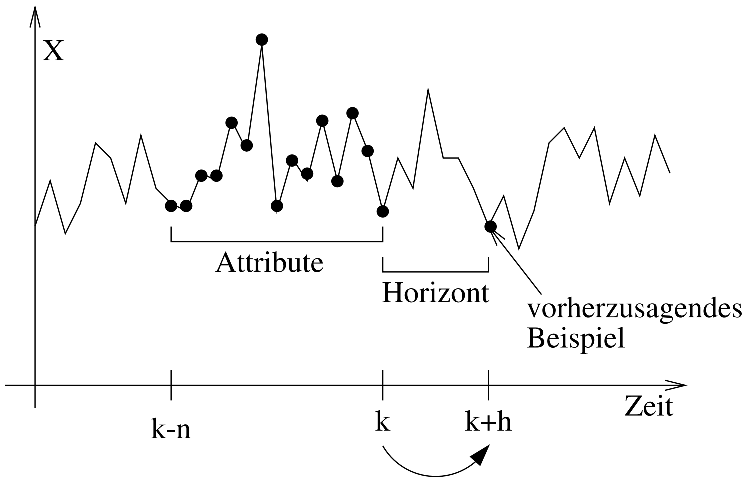


Abbildung 6.1: Aufbau der Trainingsbeispiele mit Prognosehorizont

3. der Saison, das ist eine jahreszeitlich bedingt Schwankungskomponente, die sich relativ unverändert jedes Jahr wiederholt,

4. einer Restkomponente, die die nicht zu erklärenden Einflüsse oder Störungen zusammenfasst.

Die ersten beiden Komponenten kann man zu einer einzigen Komponente, der glatten Komponente zusammenfassen. Man kann aber auch die Komponenten 2 und 3 zur zyklischen Komponente zusammenfassen. Im additiven Modell geht man davon aus, dass sich die Komponenten additiv überlagern. Bezeichnet man die glatte Komponente mit g_t, die zyklische Komponente mit z_t und die Restkomponente mit ε_t, so gilt also

$$X_t = g_t + z_t + \varepsilon_t.$$

Im hier betrachteten Fall umfasst die glatte Komponente g_t zum Beispiel einen Trend in den Daten oder die Veränderung der Zeitreihe durch ein wechselndes Konsumverhalten der Kunden oder die Veränderung des Warenangebots. Die zyklische Komponente z_t besteht aus den Einflüssen von Feiertagen und Ferien.

Für die weitere Betrachtung ist es günstig, die Existenz einer weiteren Ereignis-Komponente a_t anzunehmen. Diese Komponente fasst die Schwankungen der Zeitreihe zusammen, die direkt vom Auftreten bestimmter Ereignisse w_t und nur indirekt von der Zeit abhängen. Es gilt also $a_t = a(w_t)$. Diese Ereignisse sind typischerweise nur von kurzer Dauer, zu ihnen zählen zum Beispiel Preisänderungen oder Werbung. Das hier betrachtete Modell ist also

$$X_t = g_t + z_t + a(w_t) + \varepsilon_t.$$

Zur Prognose der Zeitreihe ist es notwendig, alle vier Komponenten gleichzeitig möglichst gut vorherzusagen. Dabei ist zu beachten, dass die Komponenten nicht unbedingt eindeutig voneinander unterscheidbar sind. Beispielsweise ist es nicht eindeutig, ob man den Einfluss eines beweglichen Feiertags wie Ostern zur zyklischen Komponente oder zur Ereignis-Komponente a_t zählt. Ebenso kann es sein, dass für einen Teil der Komponente ε_t doch noch eine Erklärung gefunden wird, die die Einordnung in eine der anderen Komponenten möglich macht.

6.1.1 Die Behandlung der Feiertage

Im Fall der Prognose von Verkaufszahlen ist ein wesentlicher Einfluss das Auftreten von Ferien und Feiertagen. Abbildung 6.2 zeigt, dass die Support Vector Machine die Zeitreihe fast im ganzen Jahr

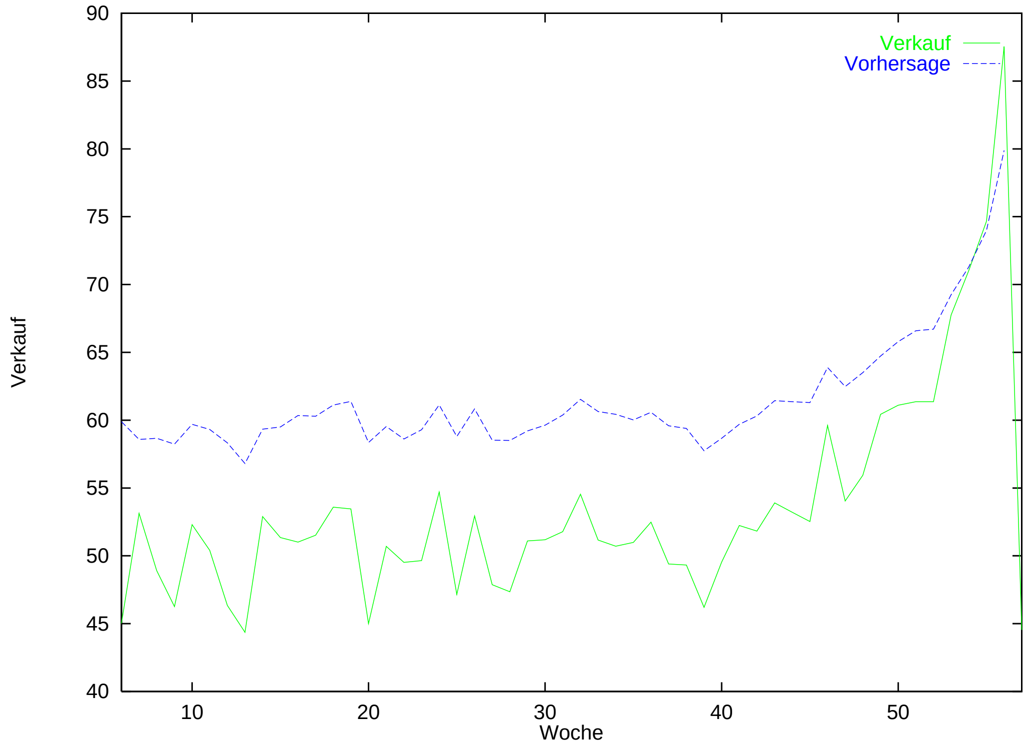


Abbildung 6.2: Vorhersage ohne Berücksichtigung von Feiertagen

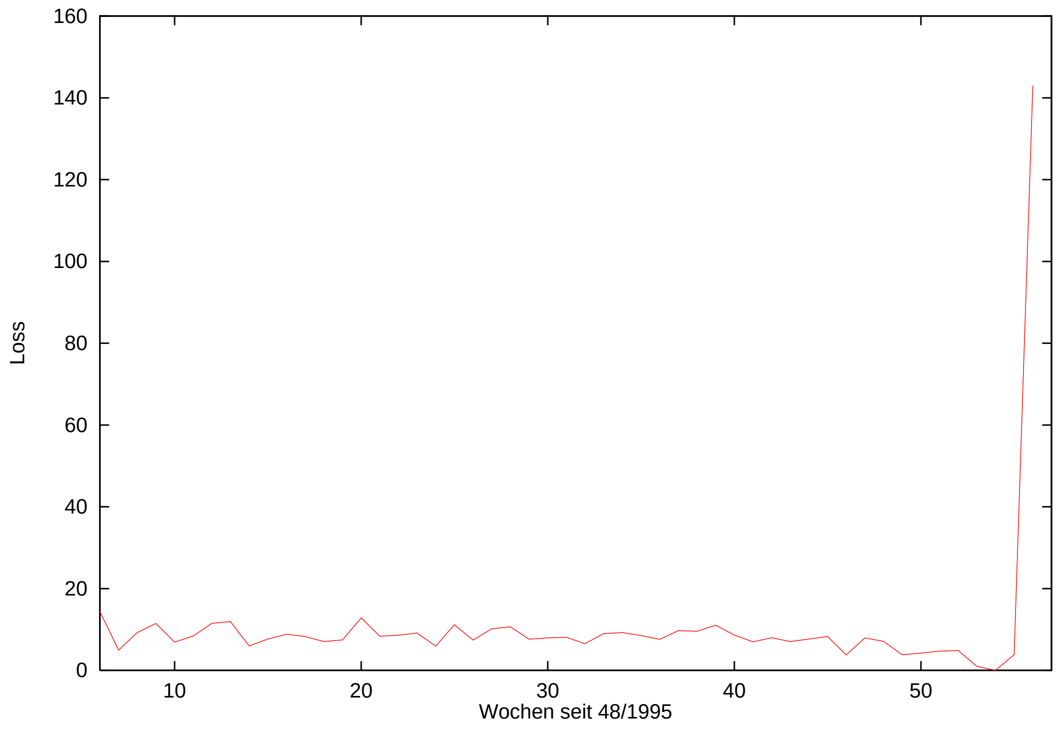


Abbildung 6.3: Loss ohne Berücksichtigung von Feiertagen

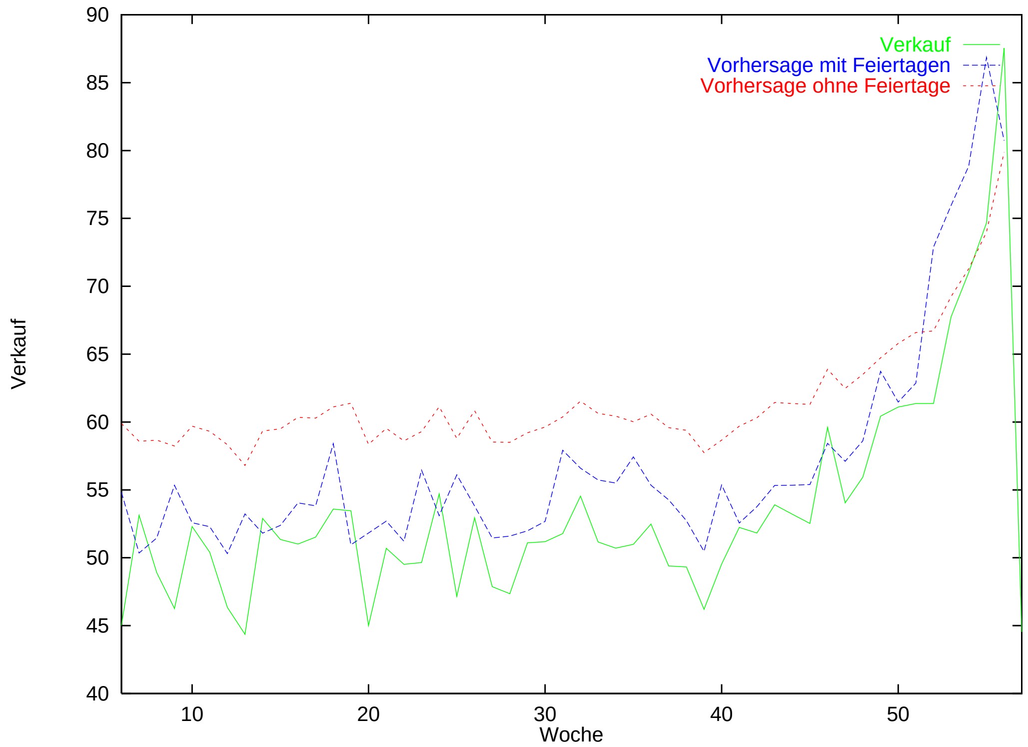


Abbildung 6.4: Vergleich der Vorhersagen mit und ohne Berücksichtigung von Feiertagen

überschätzt und nur den Anstieg zu Weihnachten unterschätzt. Der Grund dafür wird in Abbildung 6.3 klar: Dadurch, dass das Unterschätzen eines Wertes 20mal teurer ist als das überschätzen, wird zu Weihnachten ein so hoher Verlust erreicht wie das restliche Jahr über durch das Überschätzen.

Die Support Vector Machine kann also den Anstieg vor Weihnachten nicht von zufälligen Schwankungen im restlichen Jahr unterscheiden, und ist daher darauf angewiesen allgemein hohe Werte vorauszusagen, um den Gesamtverlust zu minimieren. Es liegt also auf der Hand, wie die Vorhersage verbessert werden kann: Man führt zusätzliche Binärattribute ein, die das Auftreten von Ferien oder Feiertagen anzeigen. Dabei ist es wichtig, nicht nur vor dem Auftreten von Feiertagen eine Markierung zu setzen, sondern auch danach, da nach einer Verkaufsspitze oft ein Sättigungeffekt eintritt und ein Tiefstand folgt (siehe etwa der starke Rückgang der Verkäufe nach Weihnachten in Abbildung 6.2)

Neben Weihnachten haben vermutlich auch noch folgende Feiertage und Ferien einen besonderen Einfluss auf die Verkäufe: Weihnachten, die Adventszeit, die Woche nach Weihnachten, Ostern, der Muttertag, Allerheiligen, die Sommerferien und Sommer- und Winterschlussverkauf. Benutzt man diese Attribute, so ergibt sich die Vorhersage in Abbildung 6.4 mit dem Loss aus Abbildung 6.5. Man sieht, dass die Prognose deutlich dichter an den Daten liegt und dadurch einen kleineren Verlust verursacht, obwohl sie die Daten an sechs Stellen unterschätzt.

Bei der Benutzung der Feiertage stellt sich zwangsläufig die Frage, welche Feiertage in die Prognose mit einfließen sollten. Um diese Frage zu klären, wurde einmal eine Prognose mit wenigen Feiertagsattributen, nämlich nur die Oster- und Weihnachtsattribute, und einmal eine Prognose mit vielen Feiertagsattributen, insgesamt 19 Attribute, durchgeführt.

Ein Vergleich des durchschnittlichen Verlusts bei der Vorhersage zeigt, dass die Prognose der Gruppe I sehr gering auf die Feiertagsattribute reagiert, während der Verlust der Gruppe II mit steigender Anzahl von Feiertagsattributen tatsächlich etwas sinkt. Die weiteren Versuche wurden deshalb mit allen Feiertagsattributen durchgeführt.

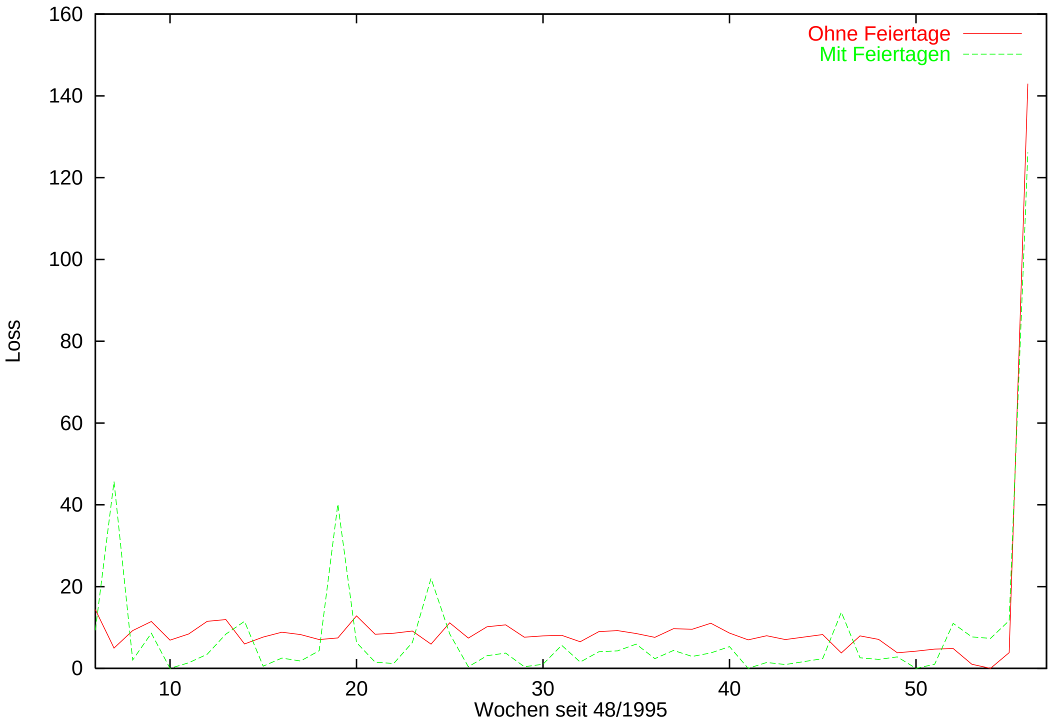


Abbildung 6.5: Vergleich des Loss mit und ohne Berücksichtigung von Feiertagen

Feiertage	Gruppe I	Gruppe II
ohne	56.445	5.563
wenige	56.446	5.545
viele	56.442	5.530

6.1.2 Die Darstellung der Zeit

Da es sich bei den betrachteten Daten um Zeitreihen handelt, also dem zeitlichen Zusammenhang der einzelnen Verkäufe eine zentrale Rolle zukommt, lohnt es sich, die Darstellung der Zeit in den Beispielen genauer zu untersuchen.

Zunächst wird die Zeitreihe $(X_1, X_2, X_3, \ldots)$ dadurch, dass für jeden Zeitpunkt k ein neues Beispiel $(a_k|X_k)$ generiert wird, auseinandergerissen. Da alle Beispiele gleichberechtigt sind und keine Reihenfolge auf den Beispiele gegeben ist, geht der zeitliche Zusammenhang verloren. Dies wird dadurch behoben, dass im Attribut a_k die letzten Verkäufe $X_{k-n}, \ldots, X_{k-1}$ berücksichtigt werden. Durch diese implizite Darstellung der Zeit ist es zum Beispiel möglich, in der Prognosefunktion den weiter zurückliegenden Verkäufen weniger Gewicht zuzuweisen als den neueren Verkäufen oder die Prognose als gewichtetes Mittel der letzten Verkäufe darzustellen.

Es ist aber auch möglich, die Zeit explizit als Attribut aufzunehmen. Dadurch wird es möglich, Änderungen in den den Daten zugrundeliegenden Prozessen nachzuvollziehen, die sich mit der Zeit ergeben. Nimmt man zum Beispiel an, dass sich die bedingte Verteilung $P(y|x)$ der Funktionswerte y bezüglich Attribute x mit der Zeit ändert, also y_t nach $P_t(y|x)$ verteilt ist, so entspricht dies der Konstruktion einer Verteilung $\hat{P}(y|(x,t))$ auf $\mathcal{X} \times \mathcal{T}$ ($\mathcal{T}$ ist der betrachtete Zeitraum). Die Support Vector Machine mit Skalarprodukt–Kernel kann dadurch zum Beispiel einen linearen Trend in den Daten erkennen: $F(\hat{x}) = F((x,t)) = w \cdot (x,t) - b = ((w_1, \ldots, w_{n-1}) \cdot x - b) + w_n \cdot t$. Mit anderen Kernelfunktionen sind auch andere Abhängigkeiten darstellbar. Diese Darstellung der Zeit wird im Folgenden *lineare Zeit* genannt.

Es ist auch denkbar, dass sich die Verteilung periodisch ändert, etwa dass im ersten Halbjahr weniger Artikel verkauft werden als im zweiten Halbjahr. Dann ist es sinnvoll ein periodisches Zeitattribut einzufügen, beispielsweise die Nummer der Woche im Jahr oder die Monatszahl. Für die Vorhersage von Verkaufszahlen bieten sich besonders die Wochen bis Weihnachten und die Wochen bis Ostern als Attribute an, da dadurch die Verkäufe in besonderem Maße beeinflusst werden. Weiter vermeidet man dadurch Probleme die entstehen, weil das Osterdatum von Jahr zu Jahr verschieden ist. In den betrachteten Daten wurde jedoch auf die Benutzung der Wochen bis Ostern verzichtet, da sich in den betrachteten Jahren (1996 und 1997) das Osterdatum nur um eine Woche unterschied.

Zeitattribut	Gruppe I	Gruppe II
ohne	56.442	5.530
periodisch	56.522	5.545
linear	56.492	5.503
periodisch und linear	56.480	5.510

Während die Prognose der Gruppe I sich durch die Zeitattribute leicht verschlechtert, ergibt sich für die Gruppe II der kleinste durchschnittliche Verlust für die lineare Zeit. Die periodische Zeit verbessert in beiden Fällen das Ergebnis nicht, was daran liegen kann, dass nur ein Jahr als Trainingsdaten benutzt werden konnte, die Support Vector Machine also nicht geeignet zwischen den linearen und periodischen Einflüssen unterscheiden konnte. Trotzdem ist der Einfluss der Zeitattribute allgemein sehr gering.

Aufgrund dieser Ergebnisse werden die weiteren Versuche mit einem linearen Zeitattribut durchgeführt.

6.1.3 Länge der Historie

Die bisherigen Versuche wurden mit den Verkäufen der letzten drei Wochen als Attribut durchgeführt. Dies basiert auf der Erkenntnis, dass sich die Verkaufszahlen im Einzelhandel sehr schnell ändern, so dass aus älteren Daten keine Informationen mehr zu holen sind. Andererseits ist es aber so, dass eine längere Historie es ermöglicht, langfristige Trends und zufällige, kurzzeitige Schwankungen besser zu erkennen. Es soll daher untersucht werden, wie sehr sich die Ergebnisse mit der Länge der Historie verändert:

Wochen	Gruppe I	Gruppe II
1	56.207	5.523
2	56.343	5.521
3	56.492	5.503
4	56.698	5.496
8	59.406	5.516

Offensichtlich genügt eine Woche bei den oft verkauften Artikeln und vier Wochen bei den selten verkauften, um eine gute Prognose zu erstellen. Weitere Wochen verschlechtern das Ergebnis wieder.

6.2 Attribute zur Identifikation weiterer Einflüsse

Um die Vorhersagequalität zu verbessern sollen Attribute konstruiert werden, mit denen andere als zeitliche Einflüsse auf die Zeitreihe identifiziert werden können. Diese Einflüsse könnten zum Beispiel Werbeaktionen und Preisänderungen sein oder allgemeine Änderungen im Kaufverhalten. Leider waren in den verfügbaren Daten keine Informationen über Werbeaktionen und Preise vorhanden, so dass der Einfluss dieser Faktoren hier nicht untersucht werden konnte. Das Erkennen und Behandeln von Einflüssen in großen Artikelgruppen wird in Kapitel 7 besprochen. Diese Kapitel beschäftigt sich mit der Approximation von Einflüssen, für die keine Erklärung aus dem Anwendungsbereich gefunden wurde und die nur aufgrund der Form der Zeitreihe erkennbar sind..

Eine interessante Frage ergibt sich aus der Darstellung der Feiertage. Bisher werden die Feiertage durch ein 0/1-Attribut behandelt. Die Support Vector Machine bestimmt dazu eine Gewichtung w, so dass im Ergebnis bei jedem Auftreten des Feiertags die Konstante w zur normalen Prognose addiert wird. Es ist aber unklar, ob dies in der Realität so eintritt. Möglich wäre auch, dass sich die Verkäufe zu einem Feiertag um einen gewissen Faktor vergrößern. Dies lässt sich mit der SVM beschreiben, indem man beim Auftreten des Feiertags nicht eine 1 sondern den vorigen Verkauf als Attribut nimmt. Trägt man sowohl eine 1 als auch den letzten Verkauf ein, wenn ein Feiertag stattfindet, so kann man damit die Abhängigkeit der Verkäufe als lineare Funktion darstellen. Der Einfluss dieser Art von Attributen wird im Folgenden untersucht:

Feiertagsattribut	Gruppe I	Gruppe II
einfach	56.492	5.503
letzter Verkauf	56.493	5.502

Die Versuche zeigen, dass die Berücksichtigung von relativen Feiertagsattributen die Prognose nicht beeinflusst. Hier wäre es sinnvoll genauer zu untersuchen wie sich das Auftreten von Feiertagen auf die Zeitreihe auswirkt.

6.2.1 Behandlung nicht erklärbarer Einflüsse

Um eine optimale Prognose zu erreichen sollte das Prognoseverfahren die in der Zeitreihe auftretenden Muster möglichst gut approximieren, ohne sich von den darüberliegenden zufälligen Schwankungen zu sehr ablenken zu lassen. Bei einer asymmetrischen Prognose ergibt sich das Problem, dass das Verfahren durch Abweichungen in die eine Richtung deutlich stärker beeinflusst wird als durch Abweichungen in die andere Richtung. Die Prognose wird also durch die zufälligen Schwankungen ungleichmässig verzerrt, wodurch die Identifikation der dahinterliegenden Muster erschwert wird.

Hat man nun keine Möglichkeit, diese Muster auf anderem Wege zu identifizieren und ein entsprechendes Attribut dafür einzuführen, so ist es denkbar die Zeitreihe zuerst durch ein symmetrisches Verfahren prognostizieren zu lassen um auf diesem Wege eine bessere Annäherung an die Zeitreihe zu bekommen. Dann kann in einem zweiten Schritt diese Prognose als Attribut des asymmetrischen Verfahrens genutzt werden um eine kostenoptimale Prognose zu erstellen.

Das symmetrische Prognoseverfahren kann beispielsweise eine einfache exponentielle Glättung sein oder wieder eine Prognose mit der Support Vector Machine. Benutzt man die Support Vector Machine so stellt sich die Frage ob man diese Prognose mit den gleichen Attributen generiert oder z.B. Attribute, die speziell im Hinblick auf eine asymmetrische Prognose erstellt wurden weglässt oder durch neue Attribute ersetzt. Zum Beispiel kann man sich überlegen einen Teil oder alle Feiertage wegzulassen, da die endgültige Prognose diese sowieso berücksichtigt.

Die folgende Tabelle zeigt die Ergebnisse der Versuche mit der symmetrischen Vorhersage der exponentiellen Glättung und der Support Vector Machine als Attribute. Für die symmetrische Prognose der SVM wurden als Attribute nicht nur die sonst verwendeten Feiertage genommen, sondern auch einmal die Feiertagsattribute weggelassen und eine Version mit wenigen Feiertagsattributen ausprobiert. Zum Vergleich ist außerdem das Ergebnis der Prognose ohne besondere Attribute angegeben.

Attribut	Gruppe I	Gruppe II
ohne	56.492	5.503
exp. Glättung	56.500	5.502
SVM, keine FT	56.710	5.436
SVM, wenige FT	56.446	5.491
SVM, viele FT	56.430	5.470

Die Qualität der Prognose kann also durch die Benutzung einer symmetrischen Prognose geringfügig verbessert werden.

6.3 Attribute zum Minimieren der zufälligen Einflüsse

Durch die Wahl einer asymmetrischen Kostenfunktion ist es notwendig, dass das Prognoseverfahren auf einen Anstieg der Werte stärker (bzw. schwächer, je nach Verlustfunktion) reagiert als auf einen Abstieg. Es ist also zunächst notwendig, dass nicht nur der letzte Wert bzw. die letzten Werte der Zeitreihe berücksichtigt, sondern auch die Veränderung dieser Werte. Weiterhin ist es notwendig, dass man zwischen steigenden und fallenden Werten unterscheiden kann. Dazu kann man neben den letzten Werten $X_{n-k}, \cdots, X_n$ auch die positiven und negativen Abweichungen $|X_n - X_{n-k}|_+, \cdots, |X_n - X_{n-1}|_+$ und $|X_n - X_{n-k}|_-, \cdots, |X_n - X_{n-1}|_-$ betrachten[2].

Betrachtet man die Zeitreihen so fällt auf, dass oft hohe Schwankungen vorhanden sind, d.h ein sehr großer Wert wird von einem kleinen gefolgt und der wiederum von einem großen. Offensichtlich ist es hier sinnlos, eine Prognose aufgrund des letzten oder vorletzten Wertes zu stellen, besser ist es die einzelnen Werte zusammenzufassen um Informationen über den Gesamtverlauf der Daten zu erhalten. Dazu kann man zum Beispiel das Maximum und das Minimum der Werte berechnen um so die Bandbreite der Werte zu bestimmen. Man kann auch die Werte jeweils aufsummieren bzw. den Mittelwert und die Varianz berechnen. Um der Asymmetrie Rechnung zu tragen ist es auch denkbar, die für Abweichungen vom Mittelwert nach oben und nach unten getrennt die Varianz zu berechnen.

Attribut	Gruppe I	Gruppe II
ohne	56.492	5.503
$\vert \times \vert_{+,-}$	56.336	5.496
max/min	56.888	5.612
Summe	56.481	5.528
Summe und $\vert$Streuung$\vert_{+,-}$	56.513	5.528

Die Versuche zeigen, dass die Beachtung der positiven und negativen Abweichung geeignet ist, die Qualität der Prognose zu verbessern. Die anderen Attribute verbessern das Ergebnis nicht.

6.4 Kernelfunktionen

Die Kernelfunktion bestimmt wesentlich die Ausdruckskraft der Support Vector Machine. Um eine gute Kernelfunktion zu finden wurden Versuche mit polynomiellen Kernels vom Grad 2 und 3, Radial-Basis-Function-Kernels mit verschiedenen Parameter n γ und Anova-Kernels vom Grad 2 und 3 gemacht. Es zeigt sich, dass neben dem linearen Kernel der Anova-Kernel eine gute Wahl ist.

[2] $|x|_+ := \max\{x, 0\}, |x|_- := \max\{-x, 0\}$

Kernel	Attribute	Gruppe I	Gruppe II
Linear	-	56.492	5.530
Polynom	$n = 2$	55.750	5.882
Polynom	$n = 3$	72.193	7.694
RBF	$\gamma = 0.1$	56.569	5.523
RBF	$\gamma = 0.01$	56.536	5.516
RBF	$\gamma = 0.001$	56.464	5.547
Anova	$n = 2, \gamma = 0.1$	56.006	5.529
Anova	$n = 2, \gamma = 0.01$	56.506	5.520
Anova	$n = 2, \gamma = 0.001$	56.572	5.529
Anova	$n = 3, \gamma = 0.01$	55.497	5.858

Eine zukünftige Arbeit könnte darin bestehen, zu versuchen einen speziellen Kernel zu konstruieren, der bereits Vorwissen über die Beziehungen der Attribute und sinnvolle Funktionenklassen untereinander beinhaltet. So ist es zum Beispiel sinnlos zu versuchen, Abhängigkeiten zwischen den Feiertagsattributen zu berücksichtigen, da die Feiertage jeweils einzeln auftreten. Stattdessen ist eine starke Abhängigkeit zwischen den einzelnen Feiertagen und den letzten Verkäufen wahrscheinlich. Eine Berücksichtigung von lokalen Korrelationen in den Daten ist zum Beispiel durch dir in [Schölkopf et al., 1997] vorgestellte Methode möglich.

6.5 Praktische Überlegungen

6.5.1 Prognosezeitraum

Da für die Berechnung einer neuen Prognose nur eine beschränkte Rechenkapazität zur Verfügung steht, ist es interessant zu untersuchen, wie lange dieselbe Hypothese verwendet werden kann bis es nötig ist, neu aus den mittlerweile gesammelten Daten zu lernen. Im Allgemeinen kann man sagen, dass eine Hypothese solange weiterverwendet werden kann, wie keine Veränderung im Verkaufsmuster des Artikel auftritt. Hat man also Informationen, dass die Verkäufe durch ein besonderes Ereignis dauerhaft verändert haben, so sollte ein neuer Lernlauf gestartet werden. Solche besonderen Ereignisse können die Einführung eines Konkurrenzprodukts, eine veränderte Plazierung des Artikels im Laden oder Veränderungen in konkurrierenden Geschäften sein.

Da Veränderungen in der Zeitreihe aber auch langsam vor sich gehen können, ohne dass ein spezielle Anlass zu erkennen ist, sollte nach einiger Zeit ein neuer Lernlauf gestartet werden. Es soll nun untersucht werden, wie sich die Qualität der Prognose mit der Zeit verändert.

Die Abbildungen 6.6 und 6.7 zeigen die Entwicklung des durchschnittlichen Loss bei der Prognose eines ganzen Jahres. Die Trainingsdaten waren die Verkaufszahlen des Jahres 1996, vorausgesagt wurden die Daten des Jahres 1997. Man sieht, dass das Loss sehr langsam ansteigt und erst nach etwa neun Monaten deutlich zu steigen beginnt. Insbesondere mit dem Einsetzen des Weihnachtsverkaufs steigt das Loss dann stark an, also sollte spätestens hier eine neue Prognosefunktion gelernt werden.

6.5.2 Prognosehorizont

Da bei der Bestellung Lieferzeiten der Produkte mit berücksichtigt werden müssen, muss die Prognose über einen längeren Zeitraum im voraus gestellt werden. Es muss also mit den Beobachtungen $X_1, \ldots, X_n$ der Wert X_{n+h} vorhergesagt werden. Dabei ist $h \geq 1$ die Länge des Prognosehorizonts.

Bei einem grösseren Prognosehorizont sind die letzten Verkäufe für die Prognose wenig aussagekräftig, da im Einzelhandel die Kunden ihre Einkäufe typischerweise nicht weit vorausplanen, sondern dann tätigen, wenn ihnen das entsprechende Produkt ausgegangen ist. Es dürfte daher wichtiger sein, langfristig vorhersehbare Einflüsse wie Feiertage oder geplante Werbeaktionen zu berücksichtigen und deren wahrscheinliche Auswirkung vorherzusagen.

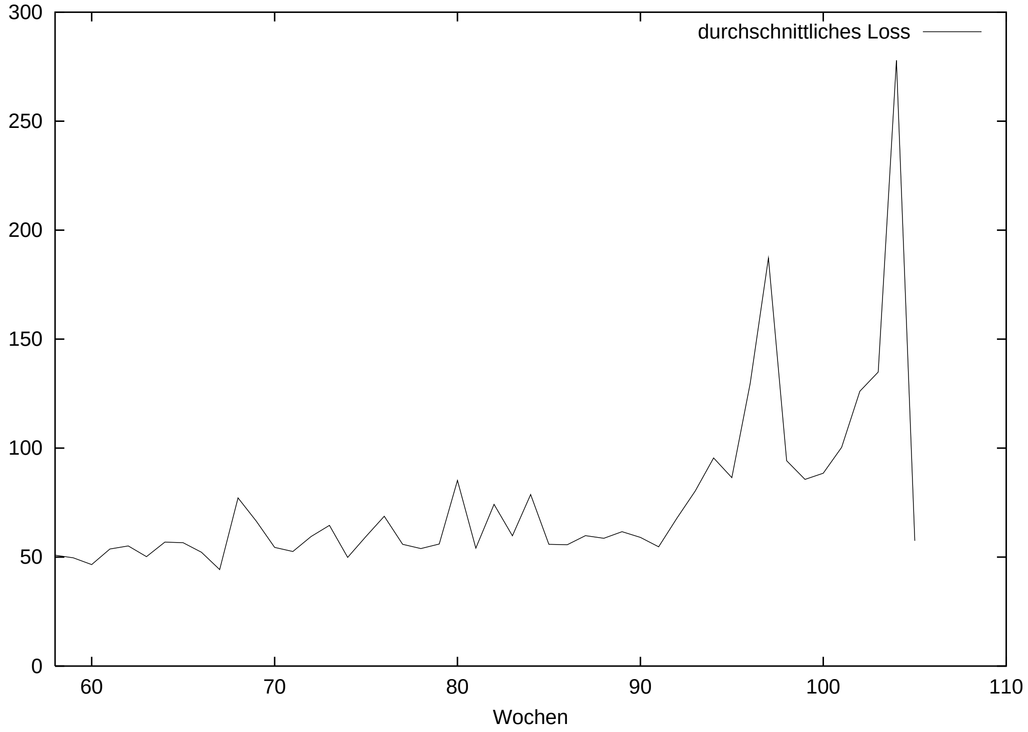


Abbildung 6.6: Loss bei Prognose über ein Jahr (Gruppe I)

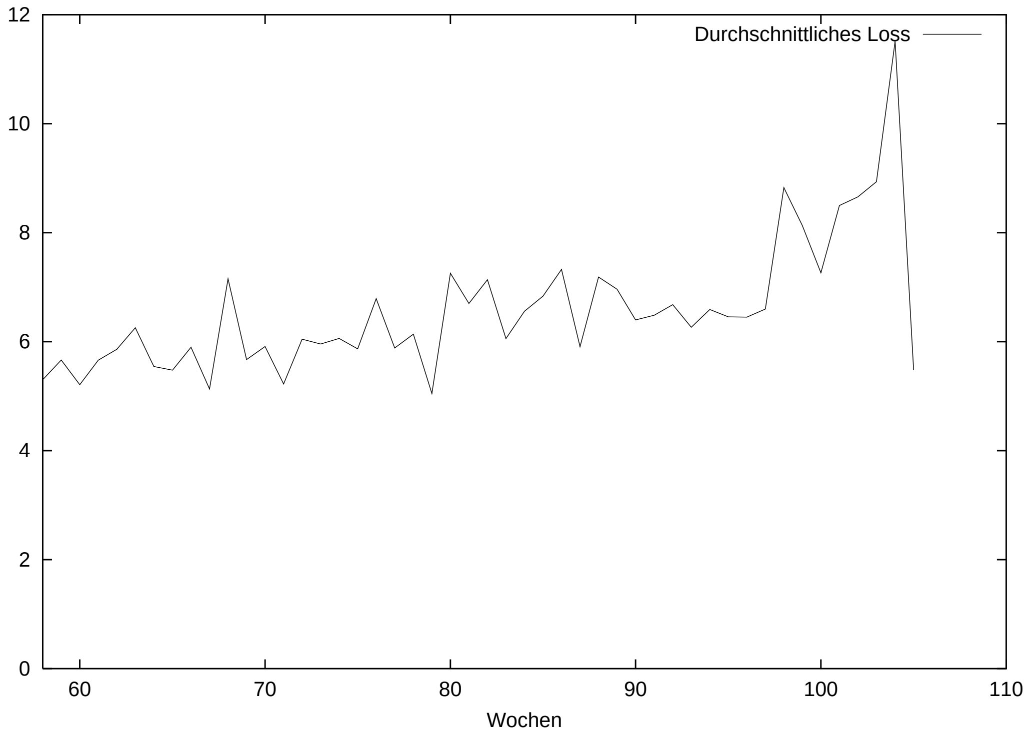


Abbildung 6.7: Loss bei Prognose über ein Jahr (Gruppe II)

Da die Prognose der Support Vector Machine auch das Auftreten von Feiertagen und Saisonfiguren berücksichtigt, sollte die Prognose hier besser ausfallen als bei Verfahren, die lediglich die letzten Werte der Zeitreihe berücksichtigen, wie etwa die exponentielle Glättung.

Horizont	Gruppe I	Gruppe II
1	56.764	5.549
2	57.044	5.543
3	57.855	5.546
4	58.670	5.543
8	60.286	5.588
13	59.475	5.605

Zum Vergleich dazu die Ergebnisse der exponentiellen Glättung:

Horizont	Gruppe I	Gruppe II
1	52.40	5.737
2	59.04	5.867
3	65.62	6.019
4	71.21	6.139
8	88.44	6.565
14	102.24	7.009

Man sieht, dass die Qualität der Prognose der Support Vector Machine sehr langsam sinkt und selbst nach einem Vierteljahr noch zu gebrauchen ist. Im Vergleich dazu sinkt die Qualität der exponentiellen Glättung schon nach kurzer Zeit stark ab.

Eine weiter Verbesserung der Prognose mit großem Horizont sollte erreichbar sein, wenn das Auftreten von Saisonfiguren in den Zeitreihen brücksichtigt wird. Dies kann geschehen, indem in einer grösseren Menge von Zeitreihen nach dem Auftreten typischer Saisonfiguren gesucht wird und der Verlauf dieser Saisonfiguren im letzten Jahr als Attribut übernommen wird.

Kapitel 7

Behandlung der dynamischen Situation im Einzelhandel

Da sich die Situation im Einzelhandel andauernd verändert ist es zweifelhaft, ob mit steigender Anzahl von Trainingsbeispielen die Genauigkeit der Prognose zu verbessern ist. Die Verkäufe eines Artikels können sich durch neue Konkurrenzprodukte oder geändertes Konsumverhalten deutlich ändern, so dass eine Prognose aufgrund der vor dieser Veränderung liegenden Daten nicht möglich ist. Weiter ist es auch nötig für neu eingeführte Artikel, d.h. ohne spezielles Vorwissen, möglichst schnell eine Prognose zu erstellen.

Dies führt zu dem Problem, den Verkauf einer Artikels aus den Verkäufen anderer Artikel und anderer Informationen zu prognostizieren. Hierzu bieten sich drei Ansätze. Ein erster Ansatz könnte darin bestehen, eine Hypothese aufzustellen, die den Verkauf einer grösseren Menge von Artikeln prognostiziert und diese Hypothese auf den gefragten Artikel anzuwenden. Aufgrund der großen Unterschiede zwischen den einzelnen Artikel und zwischen den Verkäufen eines Artikel in mehreren Filialen setzt dies aber voraus, dass bereits eine Menge von Produkten bekannt ist, die einen ähnlichen Verkaufsverlauf wie der neue Artikel aufweisen. Diese Artikelmengen können mit diversen KDD-Anwendungen gefunden werden. Die Ergebnisse der KDD-Anwendungen können dann genutzt werden, um neue Attribute für die Prognose zu konstruieren.

Ein Beispiel für eine solche KDD-Anwendung ist der Apriori-Algorithmus (siehe [Agrawal et al., 1993]) Dieser Algorithmus findet sogenannte Assoziationsregeln $A_1, \ldots, A_n \Rightarrow B$ die besagen, dass wenn ein Kunde die Artikel $A_1, \ldots, A_n$ kauft, er mit einer gewissen Wahrscheinlichkeit auch den Artikel B kauft. In diesem Fall sollten für die Prognose des Artikels B die Verkäufe der Artikel $A_1, \ldots, A_n$ berücksichtigt werden.

Der zweite Ansatz ist herauszufinden, ob sich in den Verkäufen aller Artikel wiederkehrende Muster finden lassen. Eventuell lassen sich eine oder mehrere typische Zeitreihen finden, deren Verlauf ein Großteil der Artikel folgt. Eine solche Zeitreihe könnte zum Beispiel die Verkaufszahlen von typischen Weihnachtsgeschenken sein, die kurz vor Weihnachten stark ansteigen und das restliche Jahr einen gleichmässigen, schwachen Verkauf zeigen. Dadurch vereinfacht sich das Problem der Prognose dazu, den Verkauf des neuen Artikels als geeignete Kombination der bekannten Zeitreihen darzustellen. Dieser Ansatz wird unter im Abschnitt Clustering (7.1) untersucht.

Der dritte und letzte Ansatz zur Prognose eines Artikels aus anderen Artikeln ist der, die Verkaufszahlen des neuen Artikels direkt aus Verkäufen der anderen Artikel vorherzusagen, ohne zuerst eine allgemeingültige Hypothese zu induzieren. Dieses Schlussverfahren heißt Transduktion und wird in Abschnitt 7.2 untersucht.

7.1 Clustering

Die Idee des Clustering ist, auf der Menge der betrachteten Punkte (hier: Menge der Zeitreihen) ein Abstandsmaß einzuführen und alle Punkte, die in diesem Abstandsmaß nahe genug beieinanderliegen zu einem Cluster zu vereinigen. Diese Cluster können dann z.B. durch ihren Mittelpunkt und ihren Radius beschrieben werden. In diesem Fall ist der Mittelpunkt eines Cluster wieder eine Zeitreihe, die den typischen Verlauf der Verkäufe der Artikel des Clusters beschreibt. Die Idee dabei ist, dass durch die Betrachtung des Cluster-Mittelpunkts von den zufälligen Schwankungen der Verkäufe abstrahiert und auf das gemeinsame, hinter den Zeitreihen liegende Muster geschlossen wird.

Die so identifizierten Cluster nutzt man als Attribute zum Training benutzen. Dadurch kann man herausfinden, wie sehr der betrachtete Artikel den anderen Artikeln ähnelt, und so die Verkäufe des Artikels als geeignete Kombination der typischen Zeitreihen darstellen. Um unabhängig von der Anzahl der verkauften Artikel zu sein und lediglich die Form der Zeitreihen zu betrachten ist es sinnvoll, die Zeitreihen vor dem Clustern zu normieren, etwa auf den Erwartungswert 0 und Varianz 1.

Es stellt sich nun die Frage, wie diese Informationen zur Prognose eingesetzt werden können, denn die Verkäufe der übrigen Artikel und damit die Form der typischen Zeitreihen ist zum Zeitpunkt der Prognose ebensowenig bekannt wie die Verkäufe des prognostizierten Artikels. Hier bieten sich zwei Wege.

Zum einen kann man als Attribut zur Prognose anstelle der unbekannten aktuellen Daten die letztjährigen Daten verwenden. Dies setzt voraus, dass sich die Saisonfiguren der Verkäufe wenigstens annähernd periodisch wiederholen. Dadurch ist es dann z.B. möglich, die Reaktion des Artikels auf bevorstehende Feiertage vorherzusagen. Angesichts der großen Dynamik des Einzelhandels ist es allerdings fraglich, ob man wirklich von konstanten Saisonfiguren ausgehen kann.

Die zweite Methode, die Informationen über Zeitreihencluster in die Prognose mit aufzunehmen, ist, nicht die Cluster-Mittelpunkte zu betrachten, sondern die einzelnen Artikel in Cluster selbst. Dazu führt man für jeden Cluster ein Attribut ein, dessen Wert sich als gewichtete Summe aus den letzten Verkäufen der Artikel des Clusters ergibt. Hier geht man also nicht davon aus, dass die Artikel dieselben Saisonfiguren wiederholen, sondern lediglich davon, dass die Artikel, die sich in der Vergangenheit ähnlich verkauft haben, dies auch in Zukunft tun. Man hofft also, dass sich die zufälligen Schwankungen der einzelnen Artikel gegenseitig aufheben und so das gemeinsamen Muster deutlich werden lassen. Der Nachteil dabei ist, dass jeweils nur vergangene Verkaufszahlen berücksichtigt werden können, die Reaktion der Artikel auf einen Feiertag kann beispielsweise nicht aus der Reaktion des letzten Jahres prognostiziert werden, sondern nur aus dem letztjährigen Verhalten vor dem Feiertag.

Es wurden vier Versuche mit geclusterten Daten gemacht. Im ersten Fall wurde eine Stichprobe von 2000 Artikeln aus den Zeitreihen gezogen, die nicht als Trainings- oder Testdaten benutzt wurden. Diese Zeitreihen wurden für das Jahr 1996 geclustert und die 20 grössten Cluster ausgewählt. Die daraus erhaltenen Zeitreihen wurden als Attribute für das Lernen genutzt, wobei im Jahr 1997 die Daten des Jahrs 1996 wiederholt wurden.

In den anderen drei Versuchen wurden jeweils direkt die vorherzusagenden Zeitreihen geclustert. Für jeden Cluster wurden die darin enthaltenen Zeitreihen für jede Woche aufsummiert und diese Daten für die nächste Woche zum Lernen benutzt. Die drei Versuche unterscheiden sich in der Art, wie die Daten geclustert wurden. im ersten Fall wurden die Zeitreihen direkt geclustert. Im zweiten Fall wurde jede Zeitreihe aufsummiert und in jeder Woche die Summe der Verkäufe der vorherigen Wochen als Attribut genutzt. In diesem Fall unterscheiden sich zwei Zeitreihen, die zeitlich gegeneinander verschoben sind, weniger als im ersten Fall. Im dritten Fall wurden die Zeitreihen zuerst mit einer symmetrischen Support Vector Machine vorhergesagt und anhand der Lagrange-Multiplikatoren der gelernten Funktion geclustert. Dies sollte Zeitreihen clustern, die sich auf ähnliche Art mit der Support Vector Machine vorhersagen lassen.

Leider zeigt sich, dass die Benutzung der Zeitreihen-Cluster keine Verbesserung der Prognose mit sich bringt:

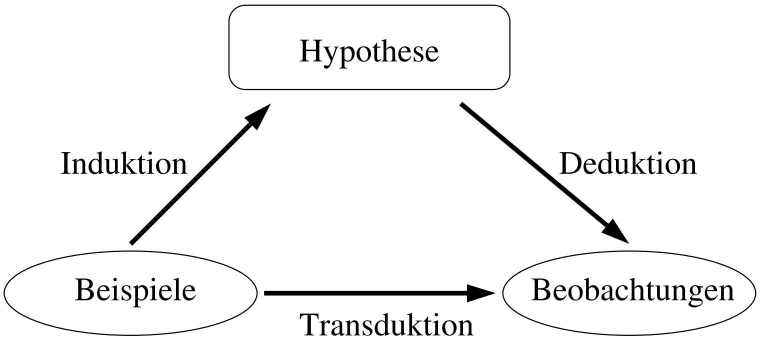


Abbildung 7.1: Prinzip des transduktiven Schlusses

Art	Gruppe I	Gruppe I
ohne	56.492	5.503
Cluster > 20	56.489	5.535
Cluster Reihe	56.547	5.512
Cluster Summe	56.583	5.510
Cluster Multiplikatoren	56.556	5.506

7.2 Transduktion

Üblicherweise beruht die Prognose von Funktionswerten oder Klassifikationen mittels des maschinellen Lernens darauf, aus einer Menge von Beispielen eine Hypothese zu induzieren, die diese Beispiele erklärt, und für neue Beobachtungen einen Funktionswert aus der Hypothese zu deduzieren. Dieses Prinzip beruht darauf, aus den Trainingsbeispielen die hinter den Trainingsdaten liegende Verteilung bzw. das logische Modell zu identifizieren, d.h. eine Hypothese zu generieren, die den aufgrund der Trainingsdaten zu erwartenden Fehler minimiert. Voraussetzung für diese Prinzip ist, dass es in der Menge aller möglichen Hypothesen eine mit dem Modell konsistente Hypothese gibt und dass diese Hypothese aus den gegebenen Beispielen eindeutig identifizierbar ist.

Angesichts der in diesem Kapitel beschriebenen Probleme der dynamischen Situation im Einzelhandel ist es zweifelhaft, ob diese Voraussetzungen noch erfüllt sind. Einerseits kann sich z.B. durch die Einführung eines Konkurrenzartikels die Form der Zeitreihe eines Artikel jederzeit verändern, so dass eine einfache Hypothese zur Erklärung der Verkäufe ausscheidet. Andererseits wäre eine Hypothese, die alle möglichen Verkäufe erklärt, falls sie überhaupt existiert, viel zu komplex, als dass man sie in der Praxis finden könnte.

Als Ausweg bietet sich hier das Schlussverfahren der Transduktion. [Zitat] Die Idee dabei ist, dass eine allgemeine Hypothese, die es erlauben würde, alle denkbaren Verkäufe des Artikels zu prognostizieren, gar nicht nötig ist um die Aufgabenstellung zu erfüllen. Es genügt eine Hypothese zu finden, die den neu zu prognostizierenden Artikel in einem gewissen Zeitintervall möglichst genau vorhersagt. Die Transduktion ist damit einfacher als die Induktion einer Hypothese und die anschließende Deduktion der neuen Prognosen, da sie nicht den Anspruch einer allgemeingültigen Hypothese hat, sondern sich lediglich auf die jeweils interessanten Daten bezieht. Die Transduktion ist aber gleichzeitig umfassender als die Suche nach einer Hypothese, die lediglich die bereits gesehenen Beispiele erklärt, da hier auch die zu prognostizierenden Daten berücksichtigt werden.

7.2.1 Transduktion bei bekannten Testdaten

Im Rahmen der Support Vector Machine wird der transduktive Schluss gewöhnlich dadurch implementiert, dass die vorherzusagenden Beobachtungen bei der Bestimmung der optimalen Hy-

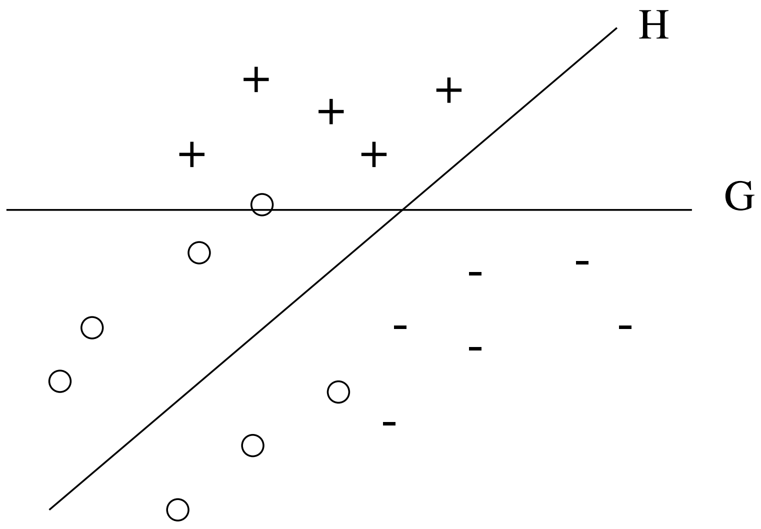


Abbildung 7.2: Transduktion bei bekannten Testdaten

perebene berücksichtigt werden, während bei der Berechnung des empirischen Fehlers natürlich nur die bekannten Klassifikationen der Trainingsmenge berücksichtigt werden. Den neuen Beobachtungen wird sozusagen eine Klassifikation derart vorgegeben, dass der insgesamt zu erwartende Fehler minimal wird. In Abbildung 7.2 könnten die positiven und negativen Beispiele auch durch die waagerechte Hyperebene G getrennt werden. Beachtet man aber auch die unklassifizierten Beobachtungen (Kreise), so zeigt sich, dass mit der diagonalen Hyperebene ein H grösserer Abstand erzielt wird.

Diese Art der Transduktion geht davon aus, dass die Trainings- und Testdaten nach derselben Verteilung unabhängig identisch verteilt gezogen wurden. Die Testdaten werden dazu genutzt, den zu erwarteten Fehler der Hypothese besser abzuschätzen als durch die Trainingsdaten allein.

7.2.2 Transduktion bei unvollständigem Vorwissen über die Testdaten

Das übliche Vorgehen der Transduktion für Support Vector Machines ist im Fall der Prognose von Zeitreihen nicht möglich, da die Beispiele unter anderem auch die letzten Werte der Zeitreihe als Attribut enthalten. Neuere Daten enthalten also immer auch die Klassifikation der Daten der direkt vorhergehenden Zeitpunkte, d.h. die Testbeobachtungen sind nur unvollständig bekannt. Es ist aber auch möglich, einen transduktiven Schluss nur aufgrund der zum Zeitpunkt der Erstellung der Prognose bekannten Attribute zu machen. Zu diesen Attributen zählen hier z.B. das Auftreten von Feiertagen oder der Zeitpunkt, zu dem die Verkäufe prognostiziert werden sollen.

Dazu lässt man die Voraussetzung, dass Trainings- und Testdaten dieselbe Verteilung besitzen, fallen, und benutzt vorhandenes Vorwissen über die Verteilung der Testdaten, um den Erwartungswert des Fehlers über den Testdaten zu minimieren. Man versucht dabei, die Trainingsbeispiele entsprechend der Verteilung der Testdaten in der Hypothese zu berücksichtigen. Dieses Verfahren ist natürlich nicht direkt durchführbar, da ja die Verteilung der Testdaten ebenfalls unbekannt ist und aus den unvollständigen Informationen über die Testdaten auch nicht geschätzt werden kann.

Unter der Annahme, dass sich die Verteilung des Trainings- und der Testdaten nicht allzu sehr unterscheidet, kann man versuchen die vorhandenen Informationen zu kombinieren. Dazu benutzt man die Heuristik, dass Trainingsbeispiele, die den Testdaten sehr ähnlich sind, in der Verteilung der Testdaten auch eine hohe Wahrscheinlichkeit haben. Die empirische Verteilung der Trainingsbeispiele wird also in Richtung der Testdaten „verschoben". In Abbildung 7.3 sei z.B. bekannt, dass die Testdaten (Kreise) einen hohen x-Wert besitzen, während über den y-Wert nichts bekannt ist. Zur Bestimmung der optimalen Hyperebene werden daher die Trainingsbeispiele mit

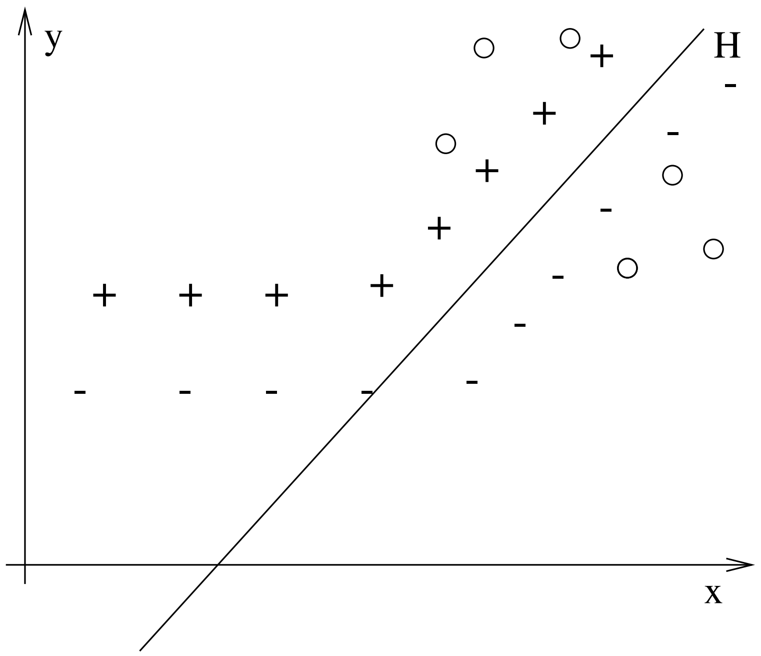


Abbildung 7.3: Transduktion bei unvollständigem Vorwissen

kleinem x-Wert ignoriert und die Hyperebene H wird nur aus den Beispielen mit hohem x-Wert generiert. Man macht also eher eine lokale Vorhersage der Daten.

Um diese Idee zu implementieren führt man auf der Menge der zum Zeitpunkt der Prognoseerstellung bekannten Attribute ein Ähnlichkeitsmaß ein und vergleicht damit die Trainingsbeispiele mit den zu prognostizierenden Daten. Dann definiert man die Fehlerfunktion so, dass Abweichungen auf den neuen Daten ähnlicheren Trainingsbeispielen strenger bestraft werden als Abweichungen auf unähnlichen Beispielen. Im Rahmen der Support Vector Machine wird das dadurch erreicht, dass jeden Trainingsbeispiel (x_i, y_i) eine Kostenkonstante $C_i \geq 0$ zugeordnet wird und jede Bedingung $0 \leq \xi_i \leq C$ ersetzt wird durch $0 \leq \xi_i \leq C_i$.

Das Ähnlichkeitsmaß sollte so gestaltet sein, dass neuere Beispiele einen höheren Wert bekommen als weiter zurückliegenden, Beispiele die genau ein Jahr zurückliegen oder am gleichen Feiertag liegen bevorzugt werden und Beispiele mit ähnlichem Saisonverlauf verstärkt berücksichtigt werden. So wird zum Beispiel vermieden, dass die Prognose von Werten im Juli durch die sehr ungewöhnlichen Daten von Weihnachten beeinflusst wird.

In diesem Fall wurde das Ähnlichkeitsmaß $\sigma(x, y)$ der Beispiele x und y aufgrund der fortlaufenden Nummer der Woche t_L, der Nummer der Woche im Jahr t_j und der Feiertagsattribute $\vec{FT}$ bestimmt. Ein einfaches Ähnlichkeitsmaß ist, den letzten Beispielen oder den Beispielen von vor einem Jahr ein höheres Gewicht zuzuordnen, zum Beispiel 5mal mehr Gewicht als den übrigen Beispielen. Eine andere Möglichkeit ist, die Ähnlichkeit der Beispiele $(t_L, t_j, \vec{FT})$ und $(\tilde{t}_L, \tilde{t}_j, \vec{FT})$ zu definieren als

$$\sigma_{rad}((t_L, t_j, \vec{FT}), (t_L^*, t_j^*, \vec{FT^*})) = C + K \cdot \exp\left(-\gamma ||(\alpha_L(t_L - t_L^*), \alpha_j(t_j - t_j^*), \alpha_{FT}(\vec{FT} - \vec{FT^*}))||^2\right)$$

mit $K \geq 0, \gamma \geq 0$ und $\alpha_L, \alpha_j, \alpha_{FT} \geq 0$ (radiales Ähnlichkeitsmaß). Eine andere Möglichkeit ist das sigmoide Ähnlichkeitsmaß

$$\sigma_{sig}((t_L, t_j, \vec{FT}), (t_L^*, t_j^*, \vec{FT^*})) = C + K \cdot \tanh\left(-\gamma ||(\alpha_L(t_L - t_L^*), \alpha_j(t_j - t_j^*), \alpha_{FT}(\vec{FT} - \vec{FT^*}))||\right)$$

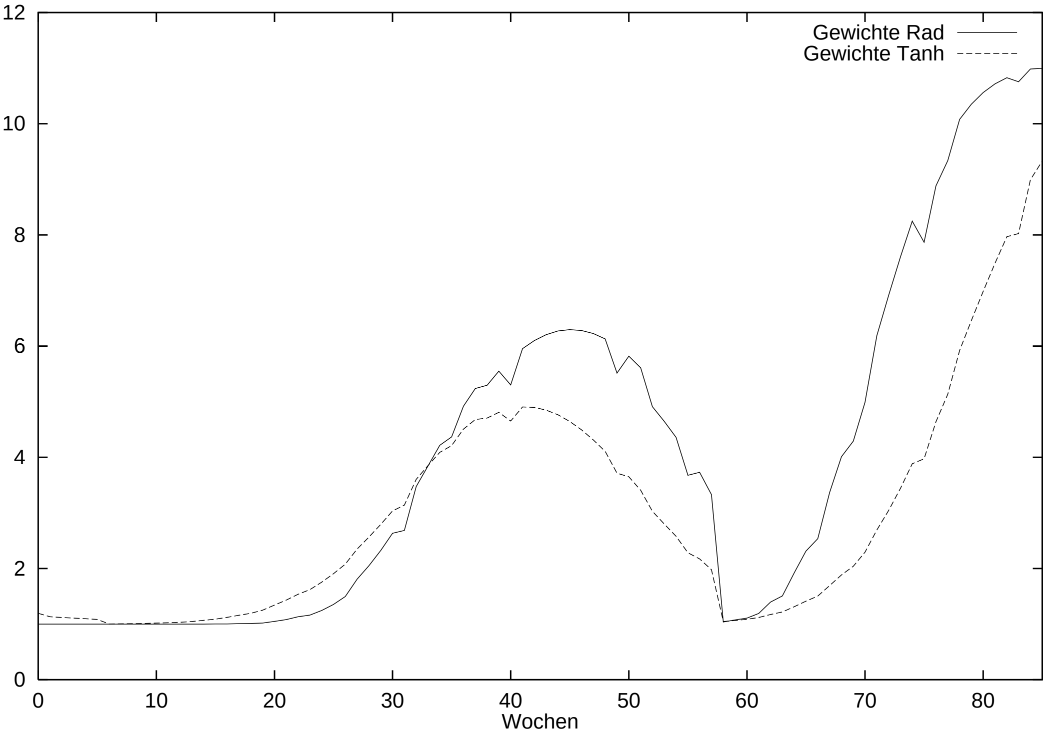


Abbildung 7.4: Gewichte der Beispiele

mit $K \geq 0, \gamma \geq 0$ und $\alpha_L, \alpha_j, \alpha_{FT} \geq 0$. Abbildung 7.4 zeigt ein Beispiel für die Wahl der Gewichte der Beispiele nach der Radial- und der Sigmoid-Methode. Die Trainingsbeispiele stammen aus den Wochen 1 bis 85 und die vorherzusagenden Beobachtungen aus den Wochen 86 bis 90. Der endgültige Wert des Gewichts eines Beispiels ist der Durchschnitt der Ähnlichkeiten über alle Testbeobachtungen, d.h. das Gewicht des Beispiels x_i bezüglich der Testmenge T ist

$$C_i = \frac{1}{|\mathrm{T}|} \sum_{j \in \mathrm{T}} \sigma(x_i, x_j).$$

Art	Gruppe I	Gruppe II
ohne	56.492	5.503
letzte Wochen $*5$	89.343	11.729
letzte Woche und Jahr $*5$	65.481	8.379
radial	55.713	5.500
sigmoid	55.510	5.688

Man sieht, dass die Transduktion für die Artikel der Gruppe I das Ergebnis verbessert. Dies sind gerade die Artikel, die ein besonders deutliches Saisonverhalten zeigen. Im Vergleich dazu sind die Verkäufe der Gruppe II viel gleichmässiger, weshalb die Prognose dieser Artikel auch nicht durch die Transduktion verbessert wird. Hier dürfte eher ausschlaggebend sein, dass man durch das Betrachten einer möglichst großen Anzahl von Beispielen das Verkaufsverhalten des Artikel gut abschätzen kann.

Kapitel 8

Zusammenfassung

In der Einleitung wurden vier wesentliche Fragen dieser Diplomarbeit formuliert. Dies waren:

- Lassen sich mit der Support Vector Machine auch asymmetrische Kostenfunktionen behandeln?
- Welche Besonderheiten hat die asymmetrische Prognose gegenüber einer symmetrischen?
- Wie lässt sich die Support Vector Machine auf das spezielle Anwendungsgebiet der Prognose von Zeitreihen in Warenwirtschaftssystemen anwenden?
- Wie lassen sich die speziellen Probleme der dynamischen Situation im Einzelhandel mit der SVM lösen?
- Hat der Einsatz der SVM Vorteile gegenüber anderen Prognoseverfahren?

Diese Fragen sollen hier noch einmal zusammenfassend beantwortet werden.

Lassen sich mit der Support Vector Machine auch asymmetrische Kostenfunktionen behandeln? In Kapitel 4 wurde gezeigt, dass das Prinzip der Support Vector Machine auch erlaubt, asymmetrische Kostenfunktionen zu behandeln. Insbesondere sind lineare und quadratische asymmetrische Kostenfunktionen ohne Verlust der Effizienz implementierbar.

Benutzt man asymmetrische Kostenfunktionen, so kann man, wie in Abschnitt 6 gezeigt wurde, spezielle Attribute konstruiert, die eine genauere Identifikation von positiven und negativen Abweichungen erlauben und so die Prognosequalität etwas verbessern.

Wie lässt sich die Support Vector Machine auf das spezielle Anwendungsgebiet der Prognose von Zeitreihen in Warenwirtschaftssystemen anwenden? Die wichtigste Eigenschaft der Support Vector Machine in diesem Anwendungsgebiet ist die Fähigkeit, die durch die Anwendung gegebene Verlustfunktion direkt in die Support Vector Machine zu integrieren und bei der Generierung der Prognose zu berücksichtigen.

Weiterhin ist die Möglichkeit von Bedeutung, eine Reihe von Attributen in der Prognose zu berücksichtigen, die alle einen gewissen Einfluss auf die Verkaufszahlen haben. Dazu zählen unter anderem Feiertage und Ferien aber auch Werbeaktionen und Preise. Auch Attribute, die aus einer Vorverarbeitung der Daten, zum Beispiel durch verschiedene KDD-Anwendungen, stammen können benutzt werden. Um diese Attribute zu benutzen ist jedoch ein genaueres Wissen darüber, welchen Einfluss diese Attribute auf die Zeitreihe haben, notwendig. Hier ist eine genauere Einsicht in den betrachteten Anwendungsfall nötig.

Zukünftige Arbeit könnte darin bestehen, die Kombination dieser Verfahren mit der Support Vector Machine genauer zu untersuchen. Ein weitere erfolgversprechender Ansatz ist, spezielle Kernelfunktionen zu implementieren, die Vorwissen über die Attribute enthalten.

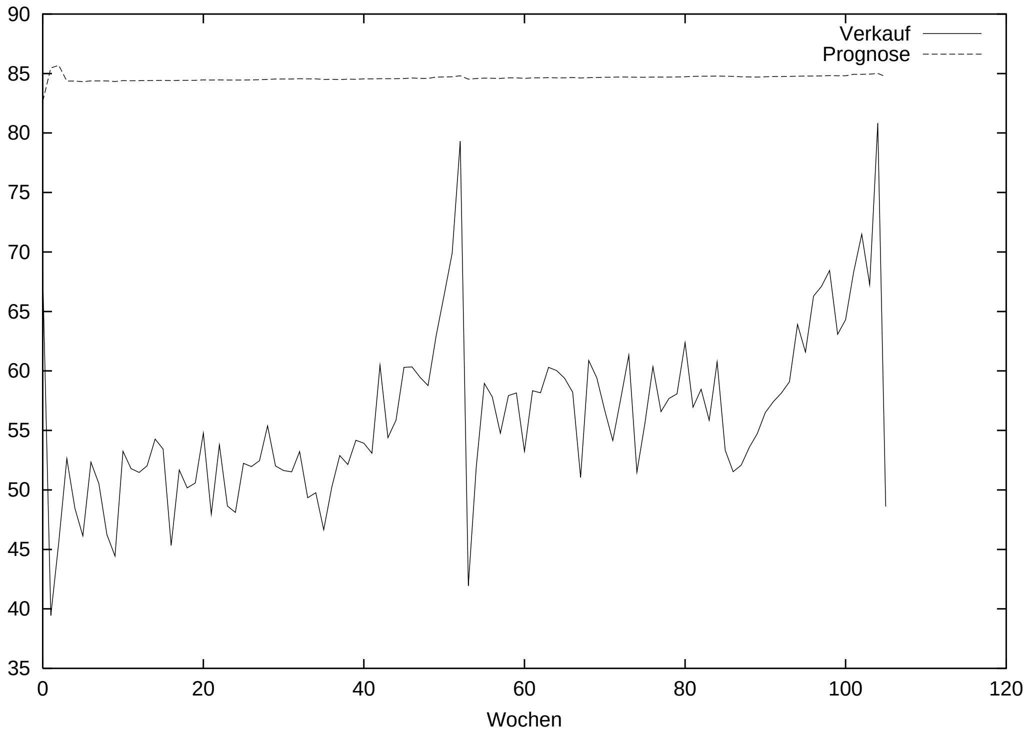


Abbildung 8.1: Durchschnittliche Prognose

Wie lassen sich die speziellen Probleme der dynamischen Situation im Einzelhandel mit der SVM lösen? Um das Problem schnell wechselnder Muster in der Zeitreihe und neuer Artikel zu lösen, wurden im Kapitel 7 das Clustering und die Transduktion vorgestellt. Hier hat sich gezeigt, dass gerade für Artikel, die ein ausgeprägtes Saisonverhalten zeigen, die Transduktion die Ergebnisse verbessern kann.

Hat der Einsatz der SVM Vorteile gegenüber anderen Prognoseverfahren? In den Kapiteln 5, 6 und 7 wurden bereits Ergebnisse der Versuche mit exponentieller Glättung und der Support Vector Machine vorgestellt. Dort zeigte sich bereits, dass die Support Vector Machine für längere Prognosehorizonte deutlich bessere Ergebnisse erzielt als die exponentielle Glättung. Bereits bei einem Prognosehorizont von zwei Wochen fällt die Qualität der Prognose der exponentiellen Glättung im Vergleich zur Support Vector Machine deutlich ab.

Dies ist gerade deshalb von Bedeutung, weil in der betriebswirtschaftlichen Praxis nicht die Vorhersage der Verkäufe der nächsten Woche aus den Daten der laufenden Woche von Bedeutung ist, sondern die Prognose über einen Wiederbeschaffungszeitraum von mehreren Wochen hinweg.

Vergleicht man die Ergebnisse der einzelnen Versuche in Kapitel 6, so stellt sich heraus, dass die einzelnen Attribute nur einen sehr geringen Einfluss auf die Qualität des Lernergebnisses hatten. Eine Erklärung dazu liefert Abbildung 8.1. Man sieht, dass die durchschnittliche Prognose der Support Vector Machine beinahe konstant ist und die Daten an allen Stellen überschätzt. Insbesondere vollzieht sie die Saisonfigur der durchschnittlichen Verkäufe nicht mit. Dies könnte daran liegen, dass die SVM durch die hohen Kosten für das Unterschätzen der Daten dazu gezwungen wird, selbst auf kleine Verkaufsspitzen stark zu reagieren und eine höhere Prognose zu stellen.

In Abbildung 8.2 sieht man, dass eine Prognose mit höherer Kapazitätskonstante C den Saisonverlauf deutlich besser nachvollzieht. Trotzdem sind die durchschnittlichen Kosten hier höher als bei der ersten Prognose. Dies liegt daran, dass die Prognose, gerade weil sie näher an den Daten liegt, die Daten öfter unterschätzt und daher höhere Kosten verursacht.

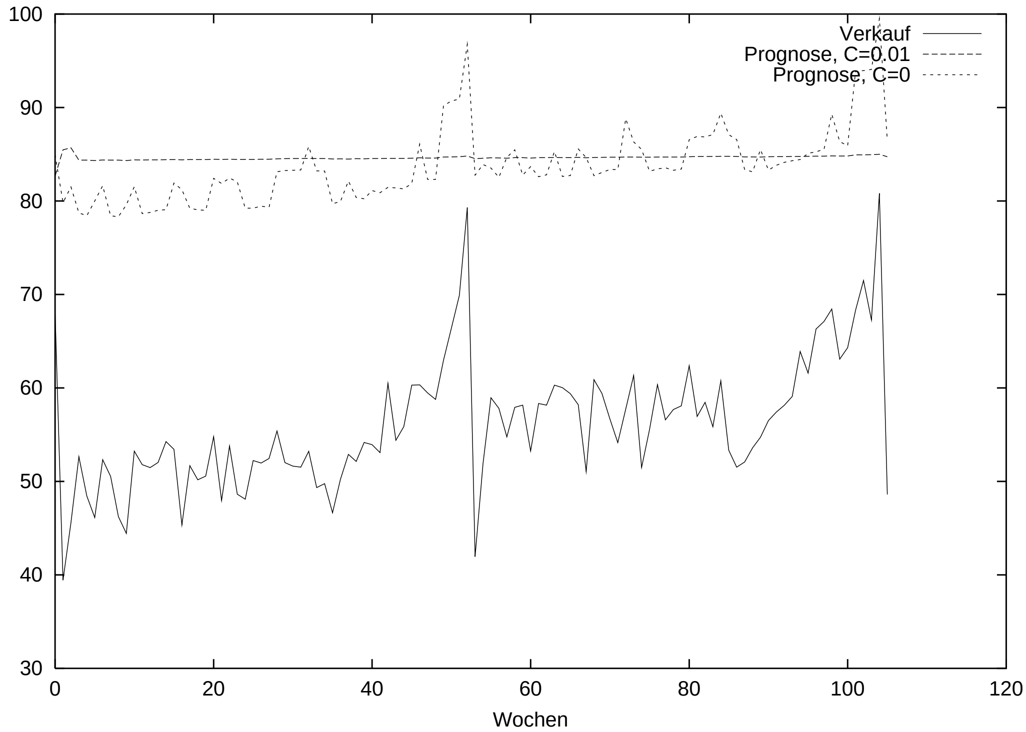


Abbildung 8.2: Vergleich der durchschnittlichen Prognose mit C=0.01 und C=1

Ein zusätzliches Problem ist dabei, dass nur sehr wenige Beispiele für die Prognose zur Verfügung stehen. Da nur die Daten eines Jahres zur Verfügung standen, ist zum Beispiel für jeden Feiertag nur ein Beispiel vorhanden. Dadurch ist es sehr schwer, den Einfluss des Feiertags auf die Verkäufe einigermassen korrekt vorherzusagen.

Literaturverzeichnis

[Agrawal et al., 1993] Agrawal, R., Imielinski, T., und Swami, A. (1993). Mining Association Rules between Sets of Items in Large Databases. In *Proceedings of the ACM SIGMOD Conference on Management of Data*, Washington, D. C.

[Agrawal et al., 1996] Agrawal, R., Mannila, H., Srikant, R., Toivonen, H., und Verkamo, A. I. (1996). Fast Discovery of Association Rules. In Fayyad, U. M., Piatetsky-Shapiro, G., Smyth, P., und Uthurusamy, R., Hrsg., *Advances in Knowledge Discovery and Data Mining*, Kapitel 12, Seiten 307–328. AAAI Press/The MIT Press, Cambridge Massachusetts, London England.

[Arminger und Götz, 1999] Arminger, G. und Götz, N. (1999). Asymmetric Loss Functions for Evaluating the Quality of Forecasts in Time Series for Goods Management Systems. SFB475–Report 22, Universität Dortmund.

[Arminger und Schneider, 1999] Arminger, G. und Schneider, C. (1999). Frequent Problems of Model Specification and Forecasting of Time Series in Goods Management Systems. SFB475–Report 21, Universität Dortmund.

[Boser et al., 1992] Boser, B. E., Guyon, I. M., und Vapnik, V. N. (1992). A Traininig Algorithm for Optimal Margin Classifiers. In Haussler, D., Hrsg., *Proceedings of the 5th Annual ACM Workshop on Computational Learning Theory*, Seiten 144–152.

[Cortes und Vapnik, 1995] Cortes, C. und Vapnik, V. N. (1995). Support–Vector Networks. *Machine Learning Journal*, 20:273–297.

[Grossmann und Terno, 1997] Grossmann, C. und Terno, J. (1997). *Numerik der Optimierung*. Teubner Studienbücher. Teubner, Stuttgart.

[Günther und Tempelmeier, 1997] Günther, H.-O. und Tempelmeier, H. (1997). *Produktion und Logistik*. Springer.

[Joachims, 1999] Joachims, T. (1999). Making large-Scale SVM Learning Practical. In Schölkopf, B., Burges, C., und Smola, A., Hrsg., *Advances in Kernel Methods - Support Vector Learning*. MIT-Press.

[Kearns und Vazirani, 1994] Kearns, M. und Vazirani, U. (1994). *An Introduction to Computational Learning Theory*. MIT Press.

[Michalski, 1986] Michalski, R. S. (1986). Understanding of the Nature of Learning: Issues and Research Directions. In Michalski, R. S., Carbonell, J. G., und Mitchell, T. M., Hrsg., *Machine Learning – An Artificial Intelligence Approach*, Jgg. 2, Kapitel 1, Seiten 3–26. Morgan Kaufmann, Palo Alto, CA.

[Mukherjee et al., 1997] Mukherjee, S., Osuna, E., und Girosi, F. (1997). Nonlinear Prediction of Chaotic Time Series using a Support Vector Machine. In *NNSP'97*.

[Müller et al., 1997] Müller, K.-R., Smola, A., Rätsch, G., Schölkopf, B., Kohlmorgen, J., und Vapnik, V. (1997). Predicting Time Series with Support Vector Machines. In *Proceedings ICANN'97*, Seite 999.

[Osuna et al., 1997] Osuna, E., Freund, R., und Girosi, F. (1997). An Improved Training Algorithm for Support Vector Machines. In Principe, J., Gile, L., Morgan, N., und Wilson, E., Hrsg., *Neural Networks for Signal Processing VII — Proceedings of the 1997 IEEE Workshop*, Seiten 276 – 285, New York. IEEE.

[Popper, 1989] Popper, K. R. (1989). *Logik der Forschung*, Kapitel VII. Mohr. 9. Auflage.

[Saunders et al., 1998] Saunders, C., Stitson, M. O., Weston, J., Bottou, L., Schölkopf, B., und Smola, A. (1998). Support Vector Machine Reference Manual. Technical report, Royal Holloway, University of London.

[Schlittgen und Streitberg, 1987] Schlittgen, R. und Streitberg, B. (1987). *Zeitreihenanalyse*. Oldenbourg Verlag.

[Schölkopf et al., 1997] Schölkopf, B., Simard, P., Smola, A., und Vapnik, V. (1997). Prior Knowledge in Support Vector Kernels. In *NIPS'97*.

[Simon, 1983] Simon, H. A. (1983). Why Should Machines Learn? In Michalski, R. S., Carbonell, J. G., und Mitchell, T. M., Hrsg., *Machine Learning — An Artificial Intelligence Approach*, Jgg. 1, Kapitel 2, Seiten 25–39. Morgan Kaufmann, Palo Alto, CA.

[Smola, 1998] Smola, A. (1998). *Learning with Kernels*. Dissertation, Technische Universität Berlin.

[Smola et al., 1996] Smola, A., Burges, C., Drucker, H., Golowich, S., van Hemmen, L., Müller, K.-R., Schölkopf, B., und Vapnik, V. (1996). Regression Estimation with Support Vector Learning Machines. Diplomarbeit, Technische Universität München.

[Smola et al., 1998] Smola, A., Schölkopf, B., und Müller, K.-R. (1998). General Cost Functions for Support Vector Regression. In Niklasson, L., Boden, M., und Ziemke, T., Hrsg., *Proceedings of the 8th International Conference on Artificial Neural Networks*.

[Smola und Schölkopf, 1998] Smola, A. J. und Schölkopf, B. (1998). A Tutorial on Support Vector Regression. Technical report, NeuroCOLT2.

[Stitson et al., 1997] Stitson, M. O., Gammerman, A., Vapnik, V., Vovk, V., Watkins, C., und Weston, J. (1997). Support Vector Regression with ANOVA Decomposition Kernels. Technical report, Royal Holloway University of London.

[Vapnik, 1998] Vapnik, V. (1998). *Statistical Learning Theory*. Wiley.

[Vapnik, 1995] Vapnik, V. N. (1995). *The Nature of Statistical Learning Theory*. Springer, New York.