Seminarausarbeitung

Bayesian Inference

Daniel Luft & Fabian Gernandt

Universität Trier

Inhaltsverzeichnis

Zusammenfassung

In diesem Paper möchten wir eine kurze Einführung in die Bayes Theorie im Rahmen des Machine Learnings geben, welche den Anspruch hat für Leser mit nur geringen Kenntnissen der Statistik und Wahrscheinlichkeitstheorie lesbar zu sein. Wir nutzen die Bayes Theorie um alternative numerische Problemstellungen zu klassischen Verfahren des Machine Learning zu erzeugen. Exemplarisch führen wir aufbauend auf der linearen Regression in die Bayesianische Regression ein, und erhalten die regularisierten Verfahren Ridge Regression und LASSO. Weiterhin greifen wir die Klassifizierungstechnik der Support Vector Machine (SVM) auf und geben eine Herleitung der sogenannten Least-Squares-SVM (LS-SVM) an. Weiterführend werden Möglichkeiten zur Hyperparameterschätzung, sowie der Selektion von Kernen in diesem Modell erarbeitet. Diese Techniken münden abschließend gemeinsam in einen Algorithmus zur Lösung der LS-SVM bei mehreren konkurrierenden Kernen.

1 Einführung in die Bayes Theorie

Das Maschinelle Lernen nimmt in dem Zeitalter der Digitalen Revolution eine zunehmend zentrale Rolle ein. Schon jetzt basieren zahlreiche Produkte von global Playern wie Google, Facebook, aber auch Kalashnikov und Co. auf Techniken des Maschinellen Lernens. Zu diesen Techniken zählen Klassifizierungsmethoden wie die Support Vector Machine (SVM) und sogenannte Neural Networks. Eines ihrer charakteristischen Merkmale ist die Verwendung großer Datensätze (Big Data), was zu interessanten Fragestellungen im Bereich der Mathematik und Statistik, aber auch der Informatik führt. Wir möchten in diesem Paper eine mathematisch-statistische Perspektive auf gewisse Problemklassen einnehmen, um hieraus einen erweiterten Zugang zu klassischen numerischen Problemstellungen des Machine Learning zu erhalten.
Ziel ist es, bestimmte Sachverhalte mit Hilfe von Modellgleichungen abzubilden. Diese Modelle bestehen in der Regel aus Inputdaten $x \in \mathbb{R}^m$, einem Output $y \in \mathbb{R}^p$, sowie von Modellparametern, welche wir mit $\theta \in \mathbb{R}^k$ bezeichnen werden. Weiterhin werden wir Datensätze $D = (d_1, \cdots, d_n), d_i \in \mathbb{R}^l$ verwenden, um unsere Modelle zu trainieren.

Hierzu wählen wir den statistischen Ansatz der sogenannten Bayes Theorie. Diese Theorie verwendet Verteilungsannahmen an Parameter θ des Modells. Solche Annahmen ermöglichen eine umfangreiche wahrscheinlichkeitstheoretische Behandlung des Modells.
Zunächst einigen wir uns auf folgende Notation von Verteilungen und Wahrscheinlichkeiten:
Sei $(\Omega, \mathcal{A}, \mathbb{P})$ ein hinreichend großer Wahrscheinlichkeitsraum, denn wir im Folgenden immer im Hintergrund voraussetzen werden. Weiterhin sei $\theta : \Omega \to \mathbb{R}^k$ eine Zufallsvariable. Dann bezeichnen wir mit $\mathbb{P}(\theta) := \mathbb{P}^\theta = \mathbb{P}(\{\theta \in \cdot\})$ das Bildmaß der Zufallsvariable θ, also Ihre Verteilung. Wir möchten hier bemerken, dass es keinen Sinn macht nach der Wahrscheinlichkeit von θ zu fragen, sondern lediglich von der Wahrscheinlichkeit dass θ bestimmte Werte annimmt. Dies ist wichtig, um Missverständnisse zu vermeiden.
Wir möchten nun in die Grundbegriffe der Bayes Theorie einführen. Die zentralen Objekte der Bayes Theorie sind die A Priori Verteilung (Prior) und die A Posteriori Verteilung (Posterior), welche durch den Satz von Bayes in Verbindung stehen.

Definition 1 (A Priori Verteilung)**.** Sei der Modellparameter θ mit Werten im $\mathbb{R}^k$ eine Zufallsvariable.
Dann heißt seine Verteilung $\mathbb{P}(\theta)$ *A Priori Verteilung*, kurz *Prior*.

Der Prior als Wahrscheinlichkeitsverteilung muss vor dem weiteren Arbeiten mit einem Bayesianischen Modell durch eine Annahme festgelegt werden. Es ist für sich nicht klar welcher Prior für ein gegebenes Modell sinnvoll ist, was zur Gefahr widersprüchlicher oder hinderlicher Annahmen führt. Deshalb ist vor dem Platzieren eines Priors eine gründliche Überlegung über Eigenschaften der Parameter wichtig. Verschiedene Priors führen zu verschiedenen Modellen und somit zu verschiedenen Problemklassen, weshalb auch eine entgegengesetzte Betrachtung möglich ist, bei der bei gegebener Problemklasse der zugrunde liegende Prior gesucht ist. Wir wählen den anderen Ansatz, und konstruieren Probleme bei gegebenem Prior.
Bevor wir die A Posteriori Verteilung einführen, erinnern wir an den Satz von Bayes.

Theorem 2 (Satz von Bayes (Bayes 1763, Laplace 1812))**.** Sei $(\Omega, \mathcal{A}, \mathbb{P})$ ein Wahrscheinlichkeitsraum, seien $A, B \in \mathcal{A}$ Ereignisse. Dann gilt für die bedingten Wahrscheinlichkeiten

$$\mathbb{P}(A|B) = \frac{\mathbb{P}(B|A)\mathbb{P}(A)}{\mathbb{P}(B)}.$$

Falls $\mathbb{P}(B) = 0$, so definieren wir den rechten Ausdruck als 0.

Beweis. Mit Definition einer bedingten Wahrscheinlichkeit erhält man

$$\frac{\mathbb{P}(B|A)\mathbb{P}(A)}{\mathbb{P}(B)} = \frac{\frac{\mathbb{P}(B\cap A)}{\mathbb{P}(A)}\mathbb{P}(A)}{\mathbb{P}(B)} = \frac{\mathbb{P}(B\cap A)}{\mathbb{P}(B)} = \mathbb{P}(A|B).$$

□

Gegeben eines Priors möchte man nun die statistischen Daten eines Modells in die Information über die Parameter miteinfliessen lassen. Die hierzu folgende Definition ist in Anlehnung an den Satz von Bayes motiviert.

Definition 3 (A Posteriori Verteilung)**.** Sei θ ein Modellparameter mit A Priori Verteilung $\mathbb{P}(\theta)$, $D = (d_1, \cdots, d_n)$ ein Datenvektor. Dann definieren wir die *A Posteriori Verteilung*, kurz *Posterior*, durch

$$\mathbb{P}(\theta|D) := \frac{\mathbb{P}(D|\theta)\mathbb{P}(\theta)}{\mathbb{P}(D)},$$

wobei $\mathbb{P}(D) = \int \mathbb{P}(D|\theta)d\mathbb{P}(\theta)$.

In dem Posterior sind die kombinierte statistische Information der Annahmen, sowie der beobachteten Daten, über θ enthalten. Der Posterior besteht aus der Likelihood $\mathbb{P}(D|\theta)$, dem Prior $\mathbb{P}(\theta)$, sowie einem Normalisierungsfaktor $\mathbb{P}(D)$. Der Prozess der Berechnung des Posteriors wird oft auch *Training* genannt, in diesem Kontext nennt man D oft auch *Trainingsdaten*. Die exakte Berechnung des Posteriors gestaltet sich bei großen Datenmengen D als schwierig, besonders wenn sogenannte latente Variablen im Modell auftreten. Latente Variablen Z sind Variablen, die nicht beobachtet werden können, und bei denen somit keine Daten zur Verfügung stehen. Diese treten häufig im Kontext von Bayesianischen Netzwerken, Modellen zur Textkategorisierung, rekurrenten Neuronalen Netzwerken und Sprach- sowie Texterkennung auf. Für einführende Beispiele von Modellen mit latenten Variablen im Machine Learning verweisen wir beispielhaft auf den Vortrag [11].
Es bestehen Möglichkeiten zur approximativen Bestimmung des Posteriors. Als Beispiel einer statistischen Methode zur approximativen Berechnung seien hier die *Markov-Chain-Monte-Carlo-Verfahren* genannt. Weiterhin gibt es analytische Approximationen, welche auf große Klassen von Verteilungen angewandt werden können. Beispielhafte Techniken sind die *Expectation-Maximization-Algorithmen* und *Variational-Bayes-Methoden*. Für eine Einführung in erstere, siehe [9], für eine ausführliche Behandlung des letzteren im Rahmen des Machine Learning siehe [10].

Wir nutzen den nun eingeführten Posterior um eine Bayesianische Art von Parameterschätzer zu definieren, den sogenannten Maximum-A-Posteriori-Schätzer. Zuvor wiederholen wir Begriff des Maximum-Likelihood-Schätzers.

Definition 4. Maximum-Likelihood-Schätzer Sei $D = (d_1, \cdots, d_n)$ ein Datensatz, θ ein $\mathbb{R}^k$-wertiger Parameter. Dann heißt der Schätzer

$$\hat{\theta}_{MLE} := \arg\max_{\theta} \mathbb{P}(D|\theta)$$

Maximum-Likelihood-Schätzer, kurz *MLE-Schätzer*.

Es sei hier erwähnt, dass ein tiefer Zusammenhang der Parameterschätzung und Optimierungsmethoden auf Mannigfaligketien besteht. Betrachtet man den Raum aller Parameter $\theta \in \Theta$ und versieht diesen mit der Fisher-Informations-Metrik, so erhält man eine riemannsche Mannigfaltigkeit, dessen Punkte als die zu den Parametern gehörigen Verteilungen interpretiert werden können. Y. Ollivier et. al. (2017) führen auf diese Weise ein Schema mit Black-Box-Ansatz zur Konstruktion von Optimierungsmethoden mit Hilfe des zur Fisher-Informations-Metrik gehörenden natürlichen Gradienten ein. Leser mit Interesse an Differentialgeometrie und numerischer Optimierung verweisen wir auf [15]. Für eine kurze Einführung in das Gebiet der Informationsgeometrie verweisen wir auf [1].
Nun führen wir in Analogie zu dem MLE-Schätzer den Bayesianischen Maximum-A-Posteriori-Schätzer ein.

Definition 5 (Maximum-A-Posteriori-Schätzer)**.** Sei $D = (d_1, \cdots, d_n)$ ein Datenvektor, θ ein $\mathbb{R}^k$-wertiger Parameter mit A Priori Verteilung $\mathbb{P}(\theta)$. Dann heißt der Schätzer

$$\hat{\theta}_{MAP} := \arg\max_{\theta} \mathbb{P}(\theta|D)$$

Maximum-A-Posteriori-Schätzer, kurz *MAP-Schätzer*. Hierbei ist $\mathbb{P}(\theta|D)$ der Posterior von θ gegeben D.

Der MAP-Schätzer ist ein statistischer Schätzer, der sich von dem MLE-Schätzer dadurch unterscheidet, dass er sowohl A Priori Annahmen an die Verteilung, als auch Daten bei der Parameterschätzung berücksichtigt. Diesen Zusammenhang erkennt man an folgender Proposition:

Proposition 6 (Darstellung des MAP-Schätzers)**.** Es gelten folgende Identitäten des MAP-Schätzers:

$$\begin{aligned}\hat{\theta}_{MAP} &:= \arg\max_{\theta} \mathbb{P}(\theta|D) \\ &= \arg\max_{\theta\in\Theta} \frac{\mathbb{P}(x|\theta)\mathbb{P}(\theta)}{\int \mathbb{P}(x|\theta)\mathrm{d}\mathbb{P}(\theta)} \\ &= \arg\max_{\theta\in\Theta} \mathbb{P}(x|\theta)\mathbb{P}(\theta) \\ &= \arg\max_{\theta\in\Theta} \log\mathbb{P}(x|\theta) + \log\mathbb{P}(\theta)\end{aligned}$$

Beweis. Definition des MAP-Schätzers und des Posteriors, sowie die Tatsache, dass $\int \mathbb{P}(x|\theta)\mathrm{d}\mathbb{P}(\theta) \in \mathbb{R}$ als Skalar nicht von θ abhängt und log monoton wachsend ist. □

Besonders anhand der letzten Identität erkennt man, dass der MAP-Schätzer die Information der Daten D im log-Likelihood-Anteil mit der A Priori-Information des log-transformierten Priors kombiniert. Diese Darstellungen werden wir im weiteren verwenden um Optimierungsprobleme. Im Wesentlichen wird es bei allen Herleitungen in den folgenden Kapiteln darum gehen, durch das Aufstellen des MAP-Schätzers in verschiedenen Situationen regularisierte oder gänzlich neue Problemstellungen und Verfahren im Machine Learning herzuleiten. Da wir nun die Grundlagen eingeführt haben, fahren wir fort mit unserem ersten nichttrivialen Anwendungsbeispiel, der Bayesianischen Regression.

2 Bayesian Regression

2.1 Multiple lineare Regression

Als erstes nicht-triviales Anwendungsbeispiel soll die Bayes Regression dienen. Dieses Gebiet ist eine Abwandlung der multiplen linearen Regression und des kleinste-Quadrate-Schätzers. Deshalb wird die klassische Problemstellung der Regression kurz wiederholt. Das grundlegende Modell der multiplen linearen Regression lautet in Matrixschreibweise

$$y \;=\; X\beta + \varepsilon.$$

Dabei ist $y \in \mathbb{R}^n$ die abhängige Variable bzw. der Modell-Output, $X \in \mathbb{R}^{n\times(K+1)}$ die Matrix der unabhängigen Variablen bzw. des Input-Datensatzes mit K Variablen, $\beta \in \mathbb{R}^{K+1}$ der Modell-Koeffizientenvektor und $\varepsilon \in \mathbb{R}^n$ der Fehlerterm. Für ε wird generell eine Normalverteilungsannahme gemacht. Die Fehlerterme sind demnach multivariat normalverteilt mit Mittelwert 0 und Varianz $\sigma^2 I_n$. Da die Störgrößen ε sowie die wahren β-Koeffizienten generell unbekannt sind, muss das Modell geschätzt werden. Der Output-Vektor mit geschätztem $\hat{\beta}$ lautet

$$\hat{y} \;=\; X\hat{\beta}.$$

Hierbei stellt sich nun die Frage nach einem geeigneten Schätzer, der das wahre β möglichst genau trifft. Der bei weitem bekannteste Schätzer für β ist der kleinste-Quadrate-Schätzer (kQ-Schätzer). Bei diesem Ansatz sollen die β-Koeffizienten des linearen Modells so bestimmt werden, dass die Summe der quadrierten Residuen minimal wird. Der Vektor der Residuen e ist dabei gegeben durch

$$e \;=\; y - \hat{y} \;=\; y - X\hat{\beta},$$

also die Abweichung vom gemessenen y zum geschätzten $\hat{y}$. Die Summe der quadrierten Residuen ist dann $e^T e$, welche als quadratische Funktion der Koeffizienten betrachtet werden kann. Es existiert somit immer ein Minimum. Dieses wird durch ableiten von $e^T e = \left(y - X\hat{\beta}\right)^T \left(y - X\hat{\beta}\right)$ nach $\hat{\beta}$ und anschließendes Nullsetzen bestimmt. Man erhält den (analytischen) kQ-Schätzer

$$\hat{\beta} \;=\; \left(X^T X\right)^{-1} X^T y.$$

Dieser besitzt besondere Eigenschaften. Nach dem Gauss-Markov Theorem (vgl. [7], S.46-47) ist der kQ-Schätzer nun derjenige Schätzer für β mit der geringsten Varianz unter allen linearen und unverzerrten Schätzern (**B**est **L**inear **U**nbiased **E**stimator, **BLUE**).
Allerdings ist der kQ-Schätzer nicht perfekt. Die Invertierung der Matrix $\left(X^T X\right)$ kann problematisch werden, da diese zwar mindestens positiv semidefinit ist, aber ein Eigenwert nahe 0 die Inverse schlecht konditioniert. Er trifft das wahre β zwar im Erwartungswert, jedoch kann dessen Varianz wegen der Invertierung unter Umständen beliebig groß und der Schätzer damit beliebig ungenau werden. Dies tritt zum Beispiel bei Multikollinearität in den Input-Daten auf, also wenn Abhängigkeiten zwischen zwei oder mehr Variablen vorliegen. Als Alternative zum kQ-Schätzer wird nun der Ridge-Schätzer betrachtet.

2.2 Ridge-Regression

Der *Ridge-Regression-Schätzer* (auch *Ridge-Schätzer*), bekannt durch *Hoerl* und *Kennard* (vgl. [2], 1970), ursprünglich von *Tikhonov* (1943), ist die wohl meist verwendete Abwandlung des kQ-Schätzers zur Regularisierung von schlecht konditionierten Problemen. Der Schätzer folgt in seiner allgemeinen Form aus der Idee, dass die Matrix $X^T X$ durch Addition einer positiv definiten Matrix sicher positiv definit wird. Konkret lautet der Ridge-Schätzer für β

$$\hat{\beta}_{Ridge} \;=\; \left(X^T X + \lambda I_k\right)^{-1} X^T y.$$

Genauer löst der Schätzer das Minimierungsproblem

$$\hat{\beta}_{Ridge} \;=\; \arg\min_{\beta} \left[(y - X\beta)^T (y - X\beta) + \lambda \beta^T \beta\right],$$

wobei der erste Summand analog zum kQ-Minimierungsproblem ist und der zweite Summand eine Nebenbedingung an die β-Koeffizienten darstellt. Diese werden durch die Nebenbedingung gegen 0 geschoben und somit betraglich kleiner gehalten, als beim kQ-Schätzer.
Obiges Zielfunktional soll nun mittels Bayes Ansatz hergeleitet werden. Die Annahmen der multiplen linearen Regression können zunächst wie folgt interpretiert werden:

$$\mathbb{P}(y_i|\beta) \;=\; \mathcal{N}(X_i\beta, \sigma^2),$$

was bedeutet, dass der Output-Vektor y i.i.d ist und als multivariat normalverteilte Zufallsvariable mit Mittelwert $X\beta$ und Kovarianzmatrix $\sigma^2 I_n$ angesehen

werden kann, was direkt aus dem Grundmodell der multiplen linearen Regression folgt.
Des weiteren wird nun eine a priori Annahme an die Regressionsgewichte β gemacht. Diese sollen nun ebenfalls normalverteilt sein. Die a priori Annahmen sind insgesamt:

(i) $\mathbb{P}(y_i|\beta) = \mathcal{N}(X_i\beta, \sigma^2)$

(ii) $\beta \sim \mathcal{N}(0, \tau^2 I_k)$

(iii) y_i und β_i i.i.d.

Mit den Annahmen folgt zum einen

$$\begin{aligned}\mathbb{P}(y|\beta) &\overset{\text{(iii)}}{=} \prod_{i=1}^{n} \mathbb{P}(y_i|\beta) \\ &\overset{\text{(i)}}{=} \prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(y_i - X_i\beta)^2}{2\sigma^2}\right),\end{aligned}$$

zum anderen

$$\begin{aligned}\mathbb{P}(\beta) &\overset{\text{(iii)}}{=} \prod_{i=1}^{k} \mathbb{P}(\beta_i) \\ &\overset{\text{(ii)}}{=} \prod_{i=1}^{k} \frac{1}{\tau\sqrt{2\pi}} \exp\left(-\frac{\beta_i^2}{2\tau^2}\right).\end{aligned}$$

Nun ergibt sich mit Proposition 6 für den MAP-Schätzer

$$\begin{aligned}
\hat{\beta}_{\text{MAP}} &= \arg\max_{\beta} \log \mathbb{P}(y|\beta) + \log \mathbb{P}(\beta) \\
&= \arg\max_{\beta} \left[\log \left(\prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(y_i - X_i\beta)^2}{2\sigma^2} \right) \right) \right. \\
&\qquad \left. + \quad \log \left(\prod_{i=1}^{k} \frac{1}{\tau\sqrt{2\pi}} \exp\left(-\frac{\beta_i^2}{2\tau^2} \right) \right) \right] \\
&= \arg\max_{\beta} \left[-\sum_{i=1}^{n} \frac{(y_i - X_i\beta)^2}{2\sigma^2} - \sum_{i=1}^{k} \frac{\beta_i^2}{2\tau^2} \right] \\
&= \arg\max_{\beta} \left[\frac{1}{2\sigma^2} \left(-\sum_{i=1}^{n} (y_i - X_i\beta)^2 - \frac{\sigma^2}{\tau^2} \sum_{i=1}^{k} \beta_i^2 \right) \right] \\
&= \arg\min_{\beta} \left[\sum_{i=1}^{n} (y_i - X_i\beta)^2 + \lambda \sum_{i=1}^{k} \beta_i^2 \right]
\end{aligned}$$

Damit erhält man das Ridge-Optimierungsproblem

$$\hat{\beta}_{Ridge} = \arg\min_{\beta} \left[(y - X\beta)^T (y - X\beta) + \lambda \beta^T \beta \right].$$

Der Ridge-Schätzer besitzt die gewünschte Eigenschaft, schon bei kleinem λ die Varianz der Modellkoeffizienten zu reduzieren, ist aber verzerrt und besitzt als Erwartungswert nicht das wahre β.
Analog dazu lassen sich weitere Schätzer herleiten. Hier soll kurz noch der Lasso-Schätzer (least absolute shrinkage and selection operator) hergeleitet werden.

2.3 Der Lasso-Schätzer

Analog zum Ridge-Schätzer lässt sich der Lasso-Schätzer über eine Verteilungsannahme an die β-Koeffizienten herleiten. Grundlage des Lasso-Schätzers ist die Überlegung, anstelle einer L^2-Regularisierung einen L^1-Term in der Nebenbedingung als Regularisierer zu verwenden. Dazu wird als Prior eine Laplace-Verteilung angesetzt. Die Dichte der Laplace-Verteilung lautet

$$f_{Laplace}(x|\mu, \tau) = \frac{1}{2\tau} \exp\left(-\frac{|x - \mu|}{\tau} \right)$$

und man wählt $\beta \sim \mathit{Laplace}(0, \tau)$ als Verteilungsannahme. Nun folgt analog zum Ridge-Schätzer mittels Proposition 6 für den MAP-Schätzer

$$\begin{aligned}
\hat{\beta}_{\text{MAP}} &= \arg\max_{\beta} \log \mathbb{P}(y|\beta) + \log \mathbb{P}(\beta) \\
&= \arg\max_{\beta} \left[\log \left(\prod_{i=1}^{n} \frac{1}{\sigma\sqrt{2\pi}} \exp\left(-\frac{(y_i - X_i\beta)^2}{2\sigma^2} \right) \right) \right. \\
&\qquad \left. + \quad \log \left(\prod_{i=1}^{k} \frac{1}{2\tau} \exp\left(-\frac{|\beta_i|}{\tau} \right) \right) \right] \\
&= \arg\max_{\beta} \left[-\sum_{i=1}^{n} \frac{(y_i - X_i\beta)^2}{2\sigma^2} - \sum_{i=1}^{k} \frac{|\beta_i|}{\tau} \right] \\
&= \arg\max_{\beta} \left[\frac{1}{2\sigma^2} \left(-\sum_{i=1}^{n} (y_i - X_i\beta)^2 - \frac{\sigma^2}{2\tau} \sum_{i=1}^{k} |\beta_i| \right) \right] \\
&= \arg\min_{\beta} \left[\sum_{i=1}^{n} (y_i - X_i\beta)^2 + \lambda \sum_{i=1}^{k} |\beta_i| \right]
\end{aligned}$$

und man erhält das Lasso-Optimierungsproblem

$$\hat{\beta}_{Lasso} = \arg\min_{\beta} \left[(y - X\beta)^T (y - X\beta) + \lambda \sum_{i=1}^{k} |\beta_i| \right].$$

Dieser Schätzer besitzt wieder andere Eigenschaften, als der Ridge-Schätzer. Die beiden größten Unterschiede zum Ridge-Schätzer liegen darin, dass der Lasso-Schätzer zwar nur algorithmisch bestimmt werden kann, dafür aber die Koeffizienten genau auf 0 setzen kann, was bei der Ridge-Regression nur im Grenzfall $\lambda \to \infty$ möglich ist.

Die beiden vorgestellten Schätzer sollen als Einführung in die Anwendung der Bayes Theorie genügen. Als nächstes wird ein weiterführendes Beispiel betrachtet, die Least-Square-Support Vector Machine als Alternative zur herkömmlichen Support Vector Machine.

3 Least-Squares-Support Vector Machine

In diesem Abschnitt soll das Bayesian Framework auf ein bekanntes Problem angewendet werden, genauer die Klassifizierung von Daten. Es wird dabei im folgenden die Binärklassifizierung betrachtet.
Generell ist bei solchen Problemen immer ein Datensatz $(x_i, y_i)_{i=1,\dots,n}$ von Features $x_i \in \mathbb{R}^{n_f}$ und Klassenzugehörigkeiten $y_i = \pm 1$ gegeben. Diese Daten werden auch Trainingsobjekte genannt. Jedes Objekt wird durch einen Vektor x_i in einem Vektorraum repräsentiert. Aufgabe der Support Vector Machine ist es nun, in diesen Raum eine Hyperebene zu finden, die als Trennfläche dient und die Trainingsobjekte in zwei Klassen teilt. Der Abstand der Vektoren, die der Hyperebene am nächsten liegen, wird dabei maximiert. Dieser breite, leere Rand soll später dafür sorgen, dass auch Objekte, die nicht genau den Trainingsobjekten entsprechen, möglichst zuverlässig klassifiziert werden.

Beim Einsetzen der Hyperebene ist es nicht notwendig, alle Trainingsvektoren zu beachten. Vektoren, die weiter von der Hyperebene entfernt liegen, beeinflussen Lage und Position der Trennebene nicht. Die Hyperebene ist nur von den ihr am nächsten liegenden Vektoren abhängig, und auch nur diese werden benötigt, um die Ebene mathematisch exakt zu beschreiben. Diese nächstliegenden Vektoren werden nach ihrer Funktion Stützvektoren (engl. support vectors) wodurch sich der Name Support Vector Machines (SVM) erklärt.

Weiterhin ist es wichtig zu beachten, dass eine Hyperebene nicht verbogen werden kann, weshalb die Daten zur Nutzung einer linearen Trennebene auch linear separierbar sein müssen. Da dies aber in der Realität nicht häufig der Fall ist, werden im Nachfolgenden die beiden Fälle der linear und nicht-linear trennbaren Trainingsdaten betrachtet.

3.1 Klassische Support Vector Machine

3.1.1 Linear trennbare Daten

Definition 7 (Hyperebene)**.** Eine Hyperebene im n-dimensionalen euklidischen Raum $\mathbb{R}^n$ ist eine Teilmenge $H \in \mathbb{R}$ der Form

$$H \;=\; \{x \in \mathbb{R}^n \mid w^T x + b \;=\; 0\},$$

wobei $w \in \mathbb{R}^n \setminus \{0\}$ ein Normalenvektor der Hyperebene ist. Die Variable b wird hierbei *Bias* genannt.

Sind zwei Klassen von Beispielen durch eine Hyperebene voneinander linear trennbar, gibt es in der Regel unendlich viele Hyperebenen, welche beide Klassen voneinander trennen. Die SVM versucht nun, von allen möglichen trennenden Hyperebenen diejenige mit minimaler quadratischer Norm $\|w\|_2^2$ auszuwählen, so dass gleichzeitig $y_i \left(w^T x_i + b\right) \geq 1$ für jedes Trainingsbeispiel x_i gilt. Dies ist äquivalent zur Maximierung des kleinsten Abstands zur Hyperebene, dem sogenannten Margin. Da hierbei keine Fehler in der Trennung zugelassen werden müssen, spricht man vor allem im Bezug zu nicht-linear trennbaren Daten auch von einem Hard-Margin. Insgesamt ergibt sich daraus der folgende Satz.

Satz 8 (Primales SVM-Problem)**.** Unter den obigen Vorüberlegungen definiert die Lösung des Minimierungsproblems

$$\begin{aligned} &\min_{w,b} \frac{1}{2}\|w\|_2^2 \\ &\text{unter } y_i \left(w^T x_i + b\right) \geq 1 \end{aligned}$$

diejenige Hyperebene, welche den Datensatz $(x_i, y_i)_{i=1,\dots,n}$ mit größtmöglichem Margin trennt. Für die namensgebenden Support Vektoren gilt hierbei Gleichheit in der Nebenbedingung.

Beweis. Siehe [8]. □

Das oben beschriebene Optimierungsproblem wird normalerweise in seiner dualen Form gelöst. Diese Formulierung ist äquivalent zu dem primalen Problem, in dem Sinne, dass alle Lösungen des dualen auch Lösungen des primalen Problems sind. Die Umrechnung ergibt sich dadurch, dass der Normalenvektor w als Linearkombination von Trainingsdaten geschrieben werden kann:

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

Die duale Form wird mit Hilfe der Lagrange-Multiplikatoren und der Karush-Kuhn-Tucker-Bedingungen hergeleitet.

Satz 9 (Duales SVM-Problem)**.** Das primale SVM-Problem ist äquivalent zum

dualen Minimierungsproblem

$$\begin{aligned} \min_{\lambda} \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y_i y_j x_i^T x_j - \sum_{i=1}^{n} \lambda_i \\ \text{unter } \sum_{i=1}^{n} \lambda_i y_i \ = \ 0, \\ 0 \leq \lambda_i. \end{aligned}$$

Beweis. Siehe [8]. □

Definition 10 (Entscheidungsfunktion)**.** Um für konkrete Daten anhand der Lösung des vorherigen Minimierungsproblems entscheiden zu können, welcher Gruppe diese angehören, verwendet man die sogenannte *Entscheidungsfunktion*

$$y \ = \ f(x) \ = \ \text{sign}\left(w^T x + b\right).$$

Je nachdem, wo sich die Datenpunkte relativ zur Hyperebene befinden (also oberhalb oder unterhalb), ergibt die Anwendung der Hyperebenengleichung einen positiven oder negativen Wert, wobei nach Anwendung der Vorzeichenfunktion nur noch ± 1 bleibt. Für Objekte, die genau auf der Trennebene liegen, wird der Wert zu 0.

3.1.2 Nicht-linear trennbare Daten

Um dem Problem der Nicht-linearen Trennbarkeit beizukommen, werden Schlupfvariablen e_i für jeden Datenpunkt x_i eingeführt. Diese sollen Fehler in der Nebenbedingung ausgleichen, wodurch Punkte im Margin selbst und sogar auf der falschen Seite der Hyperebene liegen können. Da die Verletzungen der Nebenbedingung möglichst klein gehalten werden sollen, werden die Schlupfvariablen in das Minimierungsproblem aufgenommen.

Satz 11 (Primales SVM-Problem mit Soft-Margin)**.** Unter den bei Soft Margin zulässigen Fehlern in der Trennung definiert die Lösung des Minimierungs-

problems

$$\begin{aligned} \min_{w,b} \frac{1}{2}\|w\|_2^2 + \xi \sum_{i=1}^{n} e_i \\ \text{unter } y_i \left(w^T x_i + b\right) \geq 1 - e_i, \\ 0 \leq e_i, \\ 0 < \xi \end{aligned}$$

diejenige Hyperebene, welche den Datensatz $(x_i, y_i)_{i=1,\dots,n}$ unter Zulässigkeit von Fehlern mit größtmöglichem Margin trennt.

Beweis. Siehe [8]. □

Bemerkung 12. Für die Support Vektoren gilt Gleichheit in der Nebenbedingung und $e_i = 0$. Für Datenpunkte, die innerhalb des Margin liegen, gilt $e_i > 0$ und Gleichheit in der Nebenbedingung.

Wie schon beim linear-trennbaren Datensatz wird das Minimierungsproblem in seiner dualen Form gelöst.

Satz 13 (Duales SVM-Problem mit Soft-Margin)**.** Das primale SVM-Problem mit Soft Margin ist äquivalent zum dualen Minimierungsproblem

$$\begin{aligned} \min_{\lambda} \frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \lambda_i \lambda_j y_i y_j x_i^T x_j - \sum_{i=1}^{n} \lambda_i \\ \text{unter } \sum_{i=1}^{n} \lambda_i y_i \ = \ 0, \\ 0 \leq \lambda_i \leq \xi. \end{aligned}$$

Beweis. Siehe [8]. □

3.1.3 Nicht-lineare Erweiterung mit Kernelfunktionen

Die zuvor behandelten Darstellungen der SVM klassifizieren die Daten mittels einer linearen Funktion. Diese ist jedoch nur optimal, wenn auch das zu Grunde liegende Klassifikationsproblem linear ist. In vielen Anwendungen ist dies aber nicht der Fall. Ein möglicher Ausweg ist, die Daten in einen Raum höherer Dimension abzubilden. SVMs zeichnen sich dadurch aus, dass sich diese Erweiterung elegant einbauen lässt. Dazu benutzt man Transformationsfunktionen der Form

$$\varphi \colon \mathbb{R}^{d_1} \to \mathbb{R}^{d_2}, \quad x \mapsto \varphi(x).$$

Dabei gilt $d^1 < d^2$, was die Anzahl möglicher trennender Abbildungen erhöht. Die Funktion φ ist allerdings aufwändig zu bestimmen, weshalb man den sogenannten *Kernel-Trick* verwendet. In der dualen Formulierungen des Optimierungsproblems gehen die Datenpunkte x_i nur in Skalarprodukten ein. Um den Rechenaufwand zu reduzieren ersetzt man daher $x_i^T x_j$ durch $\varphi(x_i)\varphi(x_j)$ und verwendet eine positiv definite *Kernelfunktion*

$$K(x_i, x_j) = \varphi(x_i)^T \varphi(x_j).$$

Dadurch kann eine Hyperebene in einem hochdimensionalen Raum implizit berechnet werden. Die Entscheidungsfunktion ergibt sich dabei direkt zu

$$\begin{aligned} y \;=\; f(x) \;&=\; \operatorname{sign}\left(\sum_{k=1}^{n} \lambda_k y_k \varphi(x_k)^T \varphi(x) + b\right) \\ &=\; \operatorname{sign}\left(\sum_{k=1}^{n} \lambda_k y_k K\left(x_k, x\right) + b\right), \end{aligned}$$

wenn man in Definition 10 mit $w = \sum_{k=1}^{n} \lambda_k y_k \varphi(x_k)$ substituiert.

Geeignete Kernelfunktionen sind zum Beispiel:

$$\begin{aligned} K(x,y) &= x^T y && \text{(linear)} \\ K(x,y) &= \left(x^T y + c\right)^d && \text{(polynomial)} \\ K(x,y) &= \exp\left(-\tfrac{||x-y||^2}{2\sigma^2}\right) && \text{(radial basis function, RBF)} \end{aligned}$$

Andere Kernelfunktionen und weiterführende Untersuchungen sind zu finden in [14].

3.2 Least-Squares-SVM

Wie zu Anfang des Kapitels erwähnt soll nun eine SVM mit Bayes-Framework hergeleitet werden. Dazu betrachtet man die Entscheidungsfunktion in allgemeiner Darstellung mit Kernelfunktion als Modellfunktion. Hierbei sind w und b Modellparameter, für welche nun a priori Verteilungsannahmen gemacht werden, um den MAP-Schätzer nach Proposition 6 bestimmen zu können. Zusätzlich wird ein Parameter e eingeführt, welcher nun die Abweichung der Datenpunkte von der Hyperebene bzw. vom Rand des Margin darstellt. Die Annahmen an w, b und die Abweichungen e_i lauten konkret:

(i) w ist multivariat normalverteilt mit $w \sim \mathcal{N}\left(0, \frac{1}{\mu} I_n\right)$.

(ii) b ist normalverteilt mit $b \sim \mathcal{N}\left(0, \sigma_b^2\right)$ und es gilt $\sigma_b^2 \to \infty$. Damit ist b im Grenzfall unbeschränkt gleichverteilt.

(iii) w und b sind stochastisch unabhängig.

(iv) Die Abweichungen $e_i = 1 - y_i \left(w^T \varphi\left(x_i\right) + b\right)$ sind normalverteilt mit $e_i \sim \mathcal{N}\left(0, \frac{1}{\xi}\right)$ und i.i.d.

(v) Der Datensatz D ist i.i.d.

Die Annahme (i) erfolgt dabei aus ähnlichen Überlegungen, wie bei der Ridge Regression (Kapitel 2). Die Koeffizienten von w sollen demnach nicht zu groß werden, aber auch für den Prior ergeben sich Vereinfachungen durch die Normalverteilung. Voraussetzung (ii) stellt dar, dass für b eine uniforme Annahme über die Verteilung gemacht wird, also keine Vorinformation in den Prior einfließen soll. Da eine herkömmliche Gleichverteilung jedoch beschränkt ist, verwendet man diesen Zugang.
Annahmen (iii) und (v) dienen zur Vereinfachung der a priori Verteilungen und ermöglichen erst deren Berechnung (s. u.). Die Forderung (iv) legt die Fehler e_i als Normalverteilt fest, aus ähnlichen Gründen wie bei (i). Durch die Gleichheitsbedingung wird zudem klar, dass der Fehlerterm für alle Datenobjekte, welche nicht auf dem Margin liegen, ungleich 0 ist und somit alle Datenpunkte im Minimierungsproblem betrachtet werden. Dies kann als multiple Regression in den Abweichungen e_i interpretiert werden und ist wohl der größte Unterschied zur klassischen SVM.
Die in den Forderungen auftretenden Variablen μ und ξ sind sogenannte Hyperparameter, da sie hierarchisch über den Modellparametern stehen. σ_b^2 ist ebenfalls ein Hyperparameter, fällt aber letztendlich durch den Grenzübergang weg. Mit diesen Annahmen folgt nun für den gemeinsamen Prior von w und b:

$$\begin{aligned}
\mathbb{P}\left(w, b \mid \mu, \xi, K\right) &\overset{\text{(iii)}}{=} \mathbb{P}\left(w \mid \mu, \xi, K\right) \cdot \mathbb{P}\left(b \mid \mu, \xi, K\right) \\
&\overset{\text{(i),(ii)}}{\propto} \exp\left(-\frac{\mu}{2} w^T w\right) \exp\left(-\frac{b^2}{2\sigma_b^2}\right) \\
&\overset{\sigma_b^2 \to \infty}{\longrightarrow} \exp\left(-\frac{\mu}{2} w^T w\right)
\end{aligned}$$

Weiterhin gilt:

$$\begin{aligned}\mathbb{P}(D \mid w,b,\mu,\xi,K) &\overset{\text{(v)}}{=} \prod_{i=1}^{n} \mathbb{P}(x_i,y_i \mid w,b,\mu,\xi,K)\\ &\overset{\text{(iv)}}{\propto} \prod_{i=1}^{n} \mathbb{P}(e_i \mid w,b,\mu,\xi,K)\\ &\overset{\text{(iv)}}{\propto} \exp\left(-\frac{\xi}{2}\sum_{i=1}^{n} e_i^2\right)\end{aligned}$$

Damit ergibt sich mit Proposition 6

$$\begin{aligned}\left(\hat{w}_{MAP},\hat{b}_{MAP}\right) &= \arg\max_{w,b} \mathbb{P}(w,b \mid D,\mu,\xi,K)\\ &= \arg\max_{w,b} \log\mathbb{P}(w,b \mid \mu,\xi,K) + \log\mathbb{P}(D \mid w,b,\mu,\xi,K)\\ &= \arg\max_{w,b} \left(-\frac{\mu}{2}w^Tw - \frac{\xi}{2}\sum_{i=1}^{n} e_i^2\right)\\ &= \arg\min_{w,b} \left(\frac{\mu}{2}w^Tw + \frac{\xi}{2}\sum_{i=1}^{n} e_i^2\right)\end{aligned}$$

Insgesamt erhält man den folgendem Satz.

Satz 14 (Least-Squares-SVM)**.** Betrachte ein SVM Problem mit i.i.d. Daten $D = (x_i,y_i)_{i=1,\dots,n}$. Weiterhin seien die Parameter $w \in \mathbb{R}^n, b \in \mathbb{R}$ stochastisch unabhängig und es gelten die a priori Verteilungen:

$$w \sim \mathcal{N}(0,\frac{1}{\mu}I_{n_f}) \text{ und } b \sim \mathcal{N}(0,\sigma_b^2).$$

Weiterhin sei der Fehler $e_i = 1 - y_i(w^T\phi(x_i)+b)$ normalverteilt mit $e_i \sim \mathcal{N}(0,\frac{1}{\xi})$. Dann sind die e_i i.i.d. und für $\sigma_b^2 \to \infty$ gilt:

$$\begin{aligned}\left(\hat{w}_{MAP},\hat{b}_{MAP}\right) &= \arg\min_{w,b} \frac{\mu}{2}w^Tw + \frac{\xi}{2}\sum_{i=1}^{n} e_i^2\\ &\qquad \text{unter } e_i = 1 - y_i(w^T\phi(x_i)+b), \quad i = 1,\dots,n\end{aligned}$$

Beweis. Der Beweis folgt aus den obigen Vorüberlegungen. □

Zur Lösung des Problems wird zunächst noch durch μ geteilt und mit $\gamma = \xi/\mu$ substituiert. Da n Nebenbedingungen mit Gleichheit vorliegen, kann sofort die

Lagrangefunktion aufgestellt werden. Diese lautet

$$\mathcal{L}(w, b, e; \lambda) \;=\; \frac{1}{2} w^T w + \frac{\gamma}{2} \sum_{i=1}^{n} e_i^2 - \sum_{i=1}^{n} \lambda_i \left(y_i \left(w^T \varphi(x_i) + b \right) - 1 + e_i \right).$$

Partielles differenzieren ergibt die notwendigen Optimalitätsbedingungen:

$$\begin{aligned}
&\text{(i)} && \frac{\partial \mathcal{L}}{\partial w} \;=\; 0 \quad \Leftrightarrow \quad w \;=\; \sum_{k=1}^{n} \lambda_k y_k \varphi(x_k) \\
&\text{(ii)} && \frac{\partial \mathcal{L}}{\partial b} \;=\; 0 \quad \Leftrightarrow \quad \sum_{k=1}^{n} \lambda_k y_k \;=\; 0 \\
&\text{(iii)} && \frac{\partial \mathcal{L}}{\partial e_i} \;=\; 0 \quad \Leftrightarrow \quad \lambda_i \;=\; \gamma e_i, \qquad\qquad i \;=\; 1, \ldots, n \\
&\text{(iv)} && \frac{\partial \mathcal{L}}{\partial \lambda_i} \;=\; 0 \quad \Leftrightarrow \quad 1 \;=\; y_i \left(w^T \varphi(x_i) + b \right) + e_i, \quad i \;=\; 1, \ldots, n
\end{aligned}$$

Nun werden w und e in (iv) eliminiert. Übrig bleibt das Gleichungssystem

$$\begin{aligned}
0 \;&=\; \sum_{k=1}^{n} \lambda_k y_k \\
1 \;&=\; \sum_{k=1}^{n} \left[\lambda_k y_k y_i \varphi(x_k)^T \varphi(x_i) \right] + y_i b + \gamma^{-1} \lambda_i, \quad i = 1, \ldots, n
\end{aligned}$$

In Matrixschreibweise lässt sich das LGS wie folgt darstellen:

$$\begin{bmatrix} 0 & y^T \\ y & \Omega + \gamma^{-1} I_n \end{bmatrix} \begin{bmatrix} b \\ \lambda \end{bmatrix} \;=\; \begin{bmatrix} 0 \\ 1_n \end{bmatrix},$$

wobei

$$\begin{aligned}
y \;&=\; (y_1, \ldots, y_n)^T, \\
1_n \;&=\; (1, \ldots, 1)^T \in \mathbb{R}^n, \\
\lambda \;&=\; (\lambda_1, \ldots, \lambda_n)^T, \\
\Omega_{ij} \;&=\; y_i y_j \varphi(x_i)^T \varphi(x_j) \;=\; y_i y_j K(x_i, x_j), \quad i, j = 1, \ldots, n.
\end{aligned}$$

Die Einträge von Ω sind dabei die Anwendung der Kernelfunktion K auf zwei Datenpunkte x_i und x_j multipliziert mit 1, wenn die Datenobjekte zur selben

Klasse gehören, bzw. mit -1, wenn dies nicht der Fall ist. Hierbei wird wie bei der klassischen SVM die Anwendung der schwer zu bestimmenden Transformationsfunktion φ durch die Kernelfunktion K ersetzt.
Wie zuvor bei der SVM wird hier das duale Minimierungsproblem gelöst, im Gegensatz dazu liegt hier aber ein lineares statt eines quadratischen Programms vor, welches recht einfach gelöst werden kann. Da allerdings die Umrechnung von λ nach w das anwenden der Transformationsfunktion φ erfordert, wird darauf verzichtet und die Trennebene in ihrer dualen Form dargestellt.

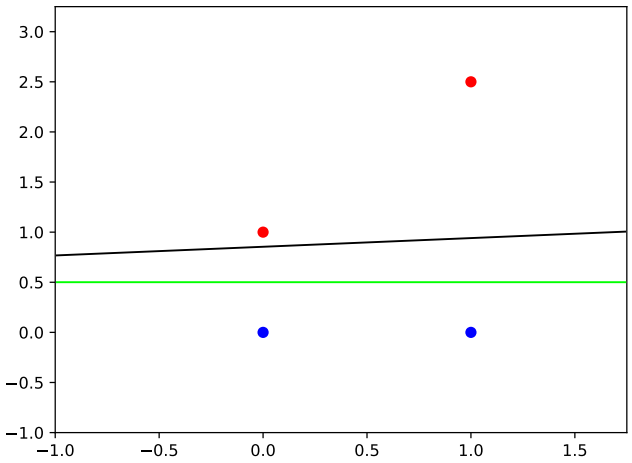

Abbildung 1: Vergleich SVM und LS-SVM: In grün eingezeichnet ist die Trennebene der SVM, in schwarz die Trennebene der LS-SVM.

In Abbildung 1 wird der Unterschied zwischen den beiden Vorgehensweisen deutlich. Die grüne Linie markiert dabei die Trennebene der SVM und ist in diesem Beispiel sehr einfach nachzuvollziehen. Der Margin soll maximal werden, was bedeutet, dass die drei unten gelegenen Punkte die Support Vektoren bilden und die Ränder des Margin und dessen Breite definieren.
Die schwarze Linie ist die Trennebene der LS-SVM und verdeutlicht den Einfluss des oberen roten Punkts auf die Hyperebene. Würden noch weitere Punkte in die rote Klasse hinzugefügt, welche weit entfernt von der blauen Klasse liegen, so würde die Trennebene weiter nach oben und eventuell über den Punkt $(0, 1)$ hinausgeschoben, so dass dieser dann fehlerhaft klassifiziert wird.
Für weitere Theorie zur LS-SVM wird auf [6] verwiesen.

4 Hyperparameterschätzung & Kernelselektion

4.1 Hyperparameterschätzung

Nachdem wir nun in dem vorangehenden Kapitel gesehen haben wie man die LS-SVM und ihr duales Problem herleitet, knüpfen wir darauf aufbauend an und leiten mögliche Arten der Hyperparameterschätzung und der Kernelselektion her.
Die Bayesianische Statistik ermöglicht eine rigorose Behandlung von fast beliebig komplexen statistischen Modellen durch A Priori Annahmen an Hyperparameter und deren Parameterabhängigkeit. Wir folgen weiterhin dem Ansatz aus [5] das Problem der Hyperparameterinferenz durch das Nutzen der so entstehenden MAP-Schätzer herzuleiten.
Zunächst betrachten wir erneut unsere LS-SVM Gleichung:

$$(\hat{w}_{MAP}, \hat{b}_{MAP}) = \arg\min_{w,b} \tfrac{\mu}{2} w^T w + \tfrac{\xi}{2} \sum_{i=1}^{n} e_i^2 \text{ unter } e_i = 1 - y_i(w^T \phi(x_i) + b)$$

Wir machen folgende Annahmen an die Verteilung der Hyperparemter $\mu, \xi > 0$:

(i) $\log(\mu) \sim \mathcal{N}(0, \sigma_\mu^2), \log(\xi) \sim \mathcal{N}(0, \sigma_\xi^2)$

(ii) $\log(\mu), \log(\xi)$ stochastisch unabhängig

(iii) $\sigma_\mu^2, \sigma_\xi^2 \to \infty$.

Die Annahmen sind dadurch zu rechtfertigen, dass als log-normalverteilte Zufallsvariablen die positiven Werte von μ und ξ respektiert werden. Der Grenzwert der Varianzen $\sigma_\mu^2, \sigma_\xi^2 \to \infty$ wird dadurch gerechtfertigt, dass man eine uniforme Annahme über die Verteilung von μ und ξ machen möchte, d.h. keine Vorinformation über ihre Werte verwenden will. Weiterhin wird die stochastische Unabhängigkeit die rechnerische Herleitung des Minimierungsproblems ermöglichen.

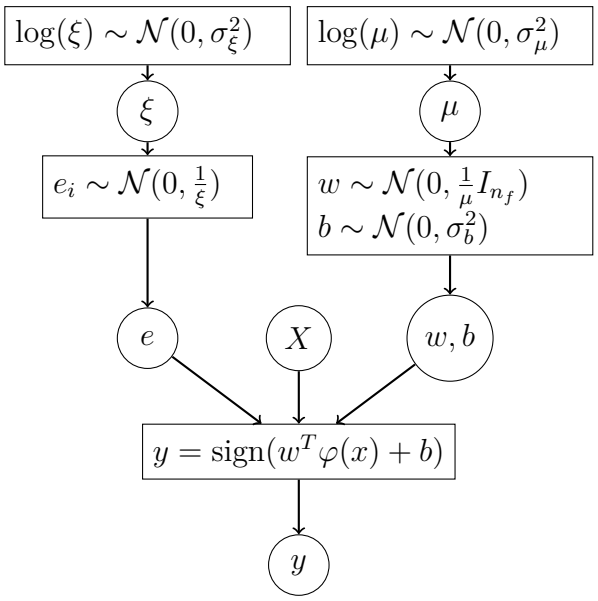

Abbildung 2: Zusammenfassung der Verteilungs- und Modellannahmen

Zunächst stellen wir über den Posterior $\mathbb{P}(\log \mu, \log \xi | D, K)$ mit Hilfe der Annahmen fest, dass

$$\begin{aligned}
&\mathbb{P}(\log(\mu), \log(\xi)|D, K) \\
&= \frac{\mathbb{P}(D|\log(\mu), \log(\xi), K)\mathbb{P}(\log(\mu), \log(\xi)|K)}{\mathbb{P}(D|K)} \\
&\propto \mathbb{P}(D|\log(\mu), \log(\xi), K) \exp(-\frac{x^2}{2\sigma_\mu^2}) \exp(-\frac{x^2}{2\sigma_\xi^2}) \\
&\to \mathbb{P}(D|\log(\mu), \log(\xi), K).
\end{aligned}$$

Somit reicht es zum Aufstellen des MAP-Schätzers die Likelihood $\mathbb{P}(D|\log(\mu), \log(\xi), K)$ gegeben der Hyperparameter μ, ξ zu betrachten. Diese Likelihood, und damit $\mathbb{P}(\log(\mu), \log(\xi)|D, K)$, lässt sich weiter mit Hilfe der MAP-Schätzer $(\hat{w}_{MAP}, \hat{b}_{MAP})$ darstellen, was wir in folgendem Lemma zusammenfassen.

Lemma 15 (Darstellung der A Postiori Wahrscheinlichkeit)**.** Es gelten die zu Beginn des Kapitels getroffenen Annahmen an μ, ξ. Dann folgt

$$\begin{aligned}\mathbb{P}(\log(\mu), \log(\xi)|D, K) &\propto \mathbb{P}(D|\log(\mu), \log(\xi), K) \\ &\propto \frac{\sqrt{\mu^{n_f}\xi^n}}{\sqrt{\det H}} \exp(-\mathcal{J}(\hat{w}_{MAP}, \hat{b}_{MAP})),\end{aligned}$$

wobei

$$\mathcal{J}(w,b) = \tfrac{\mu}{2} w^T w + \tfrac{\xi}{2} \sum_{i=1}^{n} e_i^2 \text{ und } H = \begin{pmatrix} \frac{\partial^2 \mathcal{J}}{\partial w^2} & \frac{\partial^2 \mathcal{J}}{\partial w \partial b} \\ \frac{\partial^2 \mathcal{J}}{\partial b \partial w} & \frac{\partial^2 \mathcal{J}}{\partial b^2} \end{pmatrix}.$$

Beweis. Die erste Proportionalität wurde vor Beginn des Lemmas gezeigt, für die zweite siehe [5]. □

An dieser Stelle wäre es schon möglich das Optimierungsproblem für die Hyperparameterinferenz zu formulieren, was aufgrund der Determinante $\det H$ zu keiner befriedigenden Ausdruck führt. Dem entgegnen Gestel, Suykens et. al. durch folgende Darstellung der Determinante:

Lemma 16 (Darstellung der Determinante der Hessematrix)**.** Betrache die aus dem LS-SVM Problem stammenden Ausdrücke

$$\mathcal{J}(w,b) = \tfrac{\mu}{2} w^T w + \tfrac{\xi}{2} \sum_{i=1}^{n} e_i^2 \text{ und } H = \begin{pmatrix} \frac{\partial^2 \mathcal{J}}{\partial w^2} & \frac{\partial^2 \mathcal{J}}{\partial w \partial b} \\ \frac{\partial^2 \mathcal{J}}{\partial b \partial w} & \frac{\partial^2 \mathcal{J}}{\partial b^2} \end{pmatrix}.$$

Dann gilt

$$\det(H) = n\mu^{n_f - N_{eff}} \xi \prod_{i=1}^{N_{eff}} (\mu + \xi\lambda_i),$$

wobei N_{eff} die Anzahl der Eigenwerte λ_i ungleich Null der zentrierten Matrix $M\Omega M$ mit

$$\Omega_{i,j} = K(x_i, x_j) \text{ und } M = I_n + \tfrac{1}{n} 1_v 1_v^T \text{ ist.}$$

Beweis. Siehe [5], Appendix B. □

Beide darstellenden Lemmas lassen sich nun bei einem MAP-Schätzer Ansatz kombinieren und ergeben eines der Hauptresultate zur LS-SVM:

Theorem 17 (Hyperparameterinferenz der LS-SVM)**.** Seien die Voraussetzungen an a priori Verteilungen wie zu Beginn des Kapitels.
(i) Dann sind die MAP-Schätzer der Hyperparameter μ, ξ gegeben durch

$$\begin{aligned}(\hat{\mu}_{MAP}, \hat{\xi}_{MAP}) &= \arg\min_{\mu,\xi} \; \mathcal{J}(\hat{w}_{MAP}, \hat{b}_{MAP}) + \frac{1}{2}\sum_{i=1}^{N_{eff}} \log(\mu + \xi\lambda_i) \\ &\quad - \frac{N_{eff}}{2}\log(\mu) - \frac{n-1}{2}\log(\xi) \\ &=: \arg\min_{\mu,\xi} \; \mathcal{J}_{\text{hyp}}(\mu, \xi)\end{aligned}$$

mit dem Ausgangsfunktional der LS-SVM $\mathcal{J}(w,b) = \frac{\mu}{2}w^T w + \frac{\xi}{2}\sum_{i=1}^{n} e_i^2$, sowie den nichttrivialen Eigenwerten λ_i der zentrierten Kernelmatrix.
(ii) Weiterhin sind die partiellen Ableitungen des Zielfunktionals gegeben durch

$$\begin{aligned}\frac{\partial \mathcal{J}_{\text{hyp}}}{\partial \mu} &= \hat{w}_{MAP}^T \hat{w}_{MAP} + \frac{1}{2}\sum_{i=1}^{N_{eff}} \frac{1}{\mu + \xi\lambda_i} - \frac{N_{eff}}{2\mu} \\ \frac{\partial \mathcal{J}_{\text{hyp}}}{\partial \xi} &= \sum_{i=1}^{n}(y_i - (\hat{w}_{MAP}^T \varphi(x_i) + \hat{b}_{MAP}))^2 + \sum_{i=1}^{N_{eff}} \frac{\lambda_i}{\mu + \xi\lambda_i} - \frac{N-1}{2\xi}.\end{aligned}$$

Beweis. Wir verwenden die beiden vorausgegangenen Lemmata und erhalten durch einsetzen und der Definition des MAP-Schätzers, sowie negativer log-

Transformation:

$$
\begin{aligned}
(\hat{\mu}_{MAP}, \hat{\xi}_{MAP}) &= \arg\max_{\mu,\xi} \mathbb{P}(\mu, \xi | D, K) \\
&= \arg\max_{\mu,\xi} \mathbb{P}(\log(\mu)\log(\xi) | D, K) \\
&= \arg\max_{\mu,\xi} \mathbb{P}(D | \log(\mu), \log(\xi), K) \\
&= \arg\max_{\mu,\xi} \frac{\sqrt{\mu^{n_f}\xi^n}}{\sqrt{\det H}} \exp(-\mathcal{J}(\hat{w}_{MAP}, \hat{b}_{MAP})) \\
&= \arg\max_{\mu,\xi} \frac{\sqrt{\mu^{n_f}\xi^n}}{\sqrt{n\mu^{n_f - N_{eff}}\xi \prod_{i=1}^{N_{eff}} (\mu + \xi\lambda_i)}} \exp(-\mathcal{J}(\hat{w}_{MAP}, \hat{b}_{MAP})) \\
&= \arg\max_{\mu,\xi} \frac{\sqrt{\mu^{N_{eff}}\xi^{n-1}}}{\sqrt{\prod_{i=1}^{N_{eff}} (\mu + \xi\lambda_i)}} \exp(-\mathcal{J}(\hat{w}_{MAP}, \hat{b}_{MAP})) \\
&= \arg\min_{\mu,\xi} \mathcal{J}(\hat{w}_{MAP}, \hat{b}_{MAP}) + \frac{1}{2}\sum_{i=1}^{N_{eff}} \log(\mu + \xi\lambda_i) \\
&\quad - \frac{N_{eff}}{2}\log(\mu) - \frac{n-1}{2}\log(\xi),
\end{aligned}
$$

was uns (i) liefert. (ii) erhalten wir direkt durch partielles Ableiten.

□

Dieses Minimierungsproblem ermöglicht es die Wahl der Hyperparameter mit Hilfe eines rigorosen Verfahrens zu treffen, statt durch Tuning per Hand und verschiedenen anderen heuristischen Methoden. Um das Zielfunktional aufzustellen benötigt man die Eigenwerte λ_i der zentrierten Kernelmatrix. Es fällt auf, dass die größe der zentrierten Kernelmatrix mit der Größe des Datenvektors D wächst, so dass bei großen Datenmengen extrem viele Eigenwerte auftreten können. In der Praxis wird dem begegnet, indem man approximativ die größten Eigenwerte berechnet und so das Zielfunktional annähert. Dies geschieht beispielsweise, sogar bei unendlichdimensionalem Feature-Space, mit einem Expectation-Maximization-Algorithmus, siehe etwa [12]. Für umfassende Details zur Spektraltheorie von Kernen im Rahmen des Machine Learning verweisen wir auf [13].

4.2 Kernelselektion

Nachdem wir eine mögliche Antwort auf die Frage der Auswahl von Hyperparametern gegeben haben, wollen wir uns mit der auf der nächst höheren Ebene liegenden Frage der Kernelselektion beschäftigen. Im Allgemeinen ist die Wahl des Kernels zum durchführen des Kerneltricks im Rahmen der nichtlinearen Trennung mittels LS-SVM höchst nicht-trivial. Wir verwenden, nach [5], den Ansatz die Kernel K_i und die jeweils zugehörige LS-SVM mit ihren Parametern w_i, b_i, μ_i, ξ_i als verschiedene Modelle zu betrachten, und Modellauswahl traditionell bayesianischer Art durchzuführen. Für eine umfassende Einführung in Techniken der Bayesianischen Modellselektion empfehlen wir [3]. Im folgenden Stellen wir den recht simplen Ansatz von Gestel, Suykens et. al. vor.
Ein Ansatz der Bayesianische Modelselektion basiert auf einem Ranking der Kernel durch die zugehörige A Posteriori Modellevidenz $\mathbb{P}(K_i|D)$. Wir beschränken uns hier auf die Auswahl unter endlich vielen Kernen $\{K_i\}_{i=1,\dots,m}$. Die Auswahl wird durch die Wahl des Kernels mit der größten Evidenz $\mathbb{P}(K_i|D)$ getätigt. Da wir nur endlich viele Kernel zur Auswahl stellen, läuft dies auf ein heuristisches Auswählen und sortieren hinaus.
Ähnlich wie in den vorangegangenen Betrachtungen platzieren wir A Priori Verteilungen, diesmal auf die Kernel K_I selber. Dabei nehmen wir an:

- Platziere (diskrete) a priori Verteilung auf die Kerne K_i:

$$\mathbb{P}(K_i) = p_i, \quad \sum_{i=1}^{m} \mathbb{P}(K_i) = 1$$

- Nehme eine Gleichverteilung an, d.h. $p_i = p_j$ für alle $i, j = 1, \dots, M$.
- Seien die Hyperparameter μ_i, ξ_i der jeweiligen Models mit Kern K_i stochastisch unabhängig und haben folgende Verteilung:

$$\log(\mu_i) \sim \mathcal{N}(0, \sigma_{\mu_i}^2), \quad \log(\xi_i) \sim \mathcal{N}(0, \sigma_{\xi_i}^2)$$

- $\sigma_{\mu_i}^2, \sigma_{\xi_i}^2 \to \infty$ für alle $i = 1, \dots, M$.

Es ist auch möglich ganze Familien von Kerneln zu betrachten, und unter diesen zu Selektieren. Ein Beispiel wäre die Wahl des geeigneten Kernelparameters eines RBF-Kerns. Diese Fragestellung führt jedoch zu nicht-trivialen Minimierungsproblem in dem Sinne, dass keine einfache Heuristik zu Wahl des optimalen Kernels exisitiert.

Der folgende Satz gibt als weiteres Hauptresultat an, wie man die Evidenzen $\mathbb{P}(K_i|D)$ in Proportionalität berechnen kann, und somit Kernel selektieren kann.

Theorem 18 (Berechnung der Evidenz zur Kernelselektion)**.** Seien die obigen Annahmen erfüllt. Weiterhin sei ein Datenvektor D gegeben, so dass die A Posteriori Verteilungen $\mathbb{P}(\log(\mu), \log(\xi)|D, K_i)$ eine positiv definite Kovarianzmatrix besitzt. Dann gilt:

$$\begin{aligned}\mathbb{P}(K_i|D) &\propto \mathbb{P}(D|K_i) \\ &\propto \mathbb{P}(D|\log(\hat{\mu}_{i_{MAP}}), \log(\hat{\xi}_{i_{MAP}}), K_i)\frac{\sigma_{\mu_i|D}\sigma_{\xi_i|D}}{\sigma_{\mu_i}\sigma_{\xi_i}} \\ &\propto \sqrt{\frac{\hat{\mu}_{i_{MAP}}^{N_{eff}}\hat{\xi}_{i_{MAP}}^{n-1}}{(\gamma_{eff}-1)(n-\gamma_{eff})\prod_{j=1}^{N_{eff}}(\hat{\mu}_{i_{MAP}}+\hat{\xi}_{i_{MAP}}\lambda_j)}}\end{aligned}$$

wobei $\gamma_{eff} = 1 + \sum_{j=1}^{N_{eff}} \frac{\hat{\xi}_{i_{MAP}}\lambda_j}{\hat{\mu}_{i_{MAP}}+\hat{\xi}_{i_{MAP}}\lambda_j}$.

Beweis. Die erste Proportionalität ergibt sich durch die Gleichverteilungsannahme an die K_i und Definition des Posteriors:

$$\mathbb{P}(K_i|D) = \frac{\mathbb{P}(D|K_i)\mathbb{P}(K_i)}{\mathbb{P}(D)} \propto \mathbb{P}(D|K_i)$$

Für den Rest verweisen wir auf [5]. □

Der Faktor $\frac{\sigma_{\mu_i|D}\sigma_{\xi_i|D}}{\sigma_{\mu_i}\sigma_{\xi_i}}$ wird auch mit Occam's Faktor bezeichnet, benannt nach dem mittelalterlichen Philosophen Wilhelm von Ockham, auf den auch der bekannte Begriff der Occam'schen Rasierklinge zurückgeht. Bemerkenswert an der obigen proportionalen Darstellung ist, dass sobald man MAP-Schätzer und Eigenwerte aus vorangehender Analyse gebildet hat, die Modellevidenz im Proportionalen sofort berechnet werden kann. Bei Optimierung hinsichtlich ganzer Kernelfamilien ließe sich diese Proportionalität als Ansatz nutzen um ein Minimierungsproblem herzuleiten.
Zusammenfassend münden die vorgestellten Theoreme in folgendem Algorithmus zur Selektion von Kernen bei der LS-SVM:

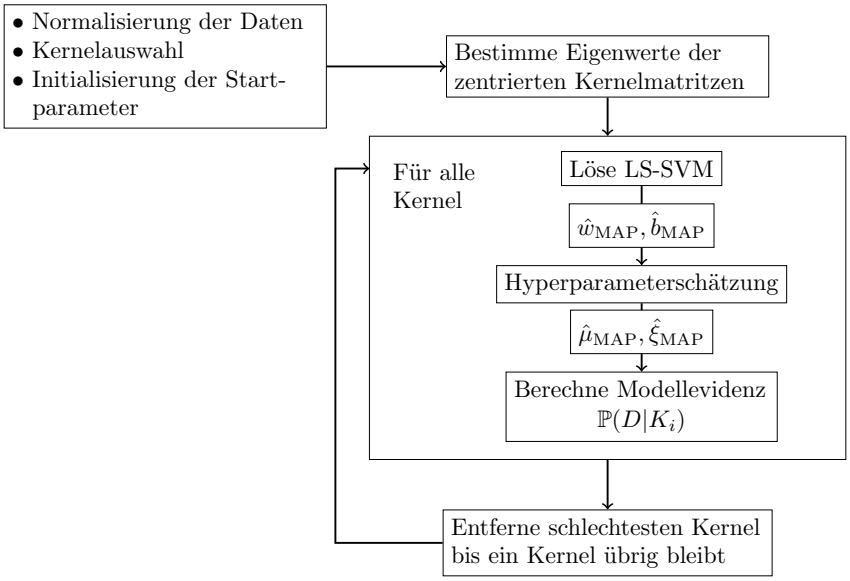

Abbildung 3: Algorithmus zur Kernelselektion bei der LS-SVM

Fazit & Ausblick

Wie wir exemplarisch gezeigt haben, lässt sich die Bayes-Theorie gewinnbringend im Kontext des Machine Learnings verwenden. Der Hauptnutzen besteht darin, numerisch umsetzbare Problemklassen rigoros herleiten zu können. Weiterhin lassen sich Probleme, wie das Hyperparametertuning und die Modellselektion, durch fundierte Ansätze statt bloßer Heuristik angehen. An diesem Punkt stellt sich uns die Frage, aus welchem Grund diese Art von Ansätzen selten in der Praxis Verwendung findet. Gründe hierfür könnten zum einen sein, dass für jedes individuelle Problem eine eigenständige Bayesianische Analyse nötig ist, was zeit- und arbeitsaufwändig ist. Zum anderen könnte es daran liegen, dass die Mischung aus Statistik und Numerik als Schnittstellenbereich zu wenig Aufmerksamkeit bekommt, und deshalb nicht verbreitet ist. Und schlussendlich kann es natürlich auch sein, dass uns nicht bekannte Nachteile bei den verwendeten Ansätzen bestehen. Im Blick auf die Zukunft des Machine Learnings wäre eine Grundlagentheorie und eine detaillierte, mathematisch saubere Untersuchung der zugrunde liegenden Prinzipien wichtig. Die Bayes-Theorie kann hierzu das mathematische Bindeglied zwischen der Statistik, der Numerik und Data Science sein. Da die Probleme und Prinzipien des Machine Learning von Natur aus ein holistisches Denken verlangen, wird eine Zusammenarbeit dieser Sparten zunehmend immer wichtiger.

Literatur

[1] University at Albany-SUNY. A. Caticha. The Basics of Information Geometry. 2014.

[2] R. W. Kennard A. E. Hoerl. Ridge regression: Biased estimation for nonorthogonal problems. Technometrics 12.1: 55-67, 1970.

[3] J. Ojanen A. Vehtari. A survey of Bayesian predictive methods for model assessment, selection and comparison. Vol. 6 142-228, 2012.

[4] T. Schmidt C. Czado. Mathematische Statistik. pages 57–63, 2011.

[5] T. Van Gestel et. al. Bayesian Framework for Least-Squares Support Vector Machine Classifiers, Gaussian Processes, and Kernel Fisher Discriminant Analysis. Vol. 14 1115-1147, 2002.

[6] T. Van Gestel et. al. Least Squares Support Vector Machines. 2002.

[7] W. Greene. Econometric Analysis. 5th ed. Englewood Cliffs, NJ: Prentice Hall, 2003.

[8] S.R. Gunn. Support vector machines for classification and regression. 1998.

[9] Universität München. M. Blume. Expectation Maximization: A Gentle Introduction. 2017.

[10] The Gatsby Computational Neuroscience Unit University College London. M. J. Beal. Variational Algorithms for approximate Bayesian Inference. 2003.

[11] C. Williams. University of Edinburgh. Latent variable models and 'deep' learning. 2011.

[12] M. Girolami R. Rosipal. An Expectation Maximization Approach to Nonlinear Component Analysis. 2001.

[13] M. L. Braun Rheinischen Friedrich-Wilhelms-Universität Bonn. Spectral Properties of the Kernel Matrix and their Relation to Kernel Methods in Machine Learning. 2005.

[14] B. Schölkopf T. Hofmann and A.J. Smola. Kernel methods in machine learning. Volume 36, Number 3, 1171-1220, 2008.

[15] L. Arnold et. al. Y. Ollivier. Information-Geometric Optimization Algorithms: A Unifying Picture via Invariance Principles. *Journal of Machine Learning Research*, 18 1-65, 2017.