TECHNISCHE
UNIVERSITÄT
DRESDEN

Vorhersage des Behandlungserfolgs mittels Deep-Learning-Verfahren zur Individualisierung der Strahlentherapie

Bachelor-Arbeit
zur Erlangung des Hochschulgrades
Bachelor of Science
im Bachelor-Studiengang Physik

vorgelegt von

Leopold Grabs
geboren am 08.10.1996 in Königs Wusterhausen

Institut für Kern- und Teilchenphysik
Fakultät Physik
Bereich Mathematik und Naturwissenschaften
Technische Universität Dresden
2018

Eingereicht am 22. Mai 2018

1. Gutachter:	Prof. Dr. Arno Straessner
2. Gutachter:	PD Dr. Steffen Löck
Betreuer:	Dipl. Inf. (FH) Stefan Leger, Dr. Alex Zwanenburg-Bezemer

Zusammenfassungen

Zusammenfassung:
In dieser Arbeit wurde untersucht, ob sich Deep-Learning-Methoden eignen, um Kopf-Hals-Tumor-Patienten in Hoch- und Niedrigrisikogruppen zur Individualisierung der Strahlentherapie einzuteilen. Anhand von 302 patientenbezogenen Ergebnis- und Bilddaten wurden statistisch relevante Prognosemodelle zur Vorhersage des Risikos für ein loko-regionäres Rezidiv (LRC) sowie der Wahrscheinlichkeit des Gesamtüberlebens konstruiert. Betrachtet wurde dabei der Zeitraum von zwei Jahren nach dem Therapiebeginn. Es wurden ein Deep-Learning-Netz (CNN) sowie konventionelle Radiomics-Modelle basierend auf Bildmerkmalen des CNNs verwendet. Bewertet wurde die Klassifizierungsgüte durch die Fläche unter der Receiver-Operating-Characteristic-Kurve (AUC). Außerdem wurden die Ergebnisse mit denen von konventionell definierten Bildmerkmalen verglichen. Das CNN sowie das beste auf CNN-Bildmerkmalen basierte konventionelle Modell zur LRC-Risiko-Vorhersage erreichten AUC-Werte von 0,62. Das konventionell berechnete, auf vordefinierten Bildmerkmalen basierende Modell, erzielte einen etwas höheren Wert von 0,65. Die Hypothese einer verbesserten Vorhersagekraft durch Verwendung von Deep-Learning-Verfahren konnte mit den gewählten Parametern somit nicht bestätigt werden. Durch die Reduktion von beobachteten Überanpassungseffekten der Modelle könnten zukünftig jedoch bessere Ergebnisse erzielt werden.

Abstract:
The prediction of clinical outcome of head and neck cancer patients for treatment individualisation is an important issue in radiotherapy. The performance of prognostic models based on deep learning methods was investigated in this coursework. The goal was to predict the risk of loco-regional tumour control (LRC) and the probability of overall survival. Therefore time-to-event data of 302 patients were used. A convolutional neuronal network (CNN) and several deep features based conventional radiomics models were built to classify patients in high and low risk groups. Model performance was assessed by the area under the receiver operating characteristic curve (AUC). Moreover the results were compared to those obtained from coventionally defined features. Both, the CNN itself as well as the deep features based radiomics models achieved results of $AUC = 0.62$ in LRC prediction. The conventionally built model revealed a slightly higher AUC value of 0.65. Hence, the hypothesis of an improved classification performance of models based on deep learning methods could not be confirmed with the investigated parameters. Nevertheless, the results might be improved in future studies by reducing the observed overfitting.

Inhaltsverzeichnis

Abbildungsverzeichnis

Tabellenverzeichnis

1 Einleitung

Im Jahr 2012 wurden weltweit 14,1 Millionen neue Krebspatienten sowie 8,2 Millionen durch Krebs verursachte Todesfälle registriert [13]. Etwa jeder vierte Todesfall in Deutschland wurde im Jahr 2014 durch Krebs verursacht und es wird davon ausgegangen, dass aufgrund der steigenden Lebenserwartung die Zahl der Neuerkrankungen in den kommenden Jahren sogar noch steigen wird [6]. Speziell an Kopf-Hals-Tumoren erkranken in Deutschland jährlich ca. 9350 Männer und 3740 Frauen [14]. Es besteht die Notwendigkeit, die bestehenden Therapieoptionen wie Operation, Chemotherapie und Strahlentherapie weiter zu verbessern.

Die Strahlentherapie beruht auf der Verwendung von ionisierender Strahlung, um die Krebszellen abzutöten bzw. deren Teilung zu verhindern. Auch bei gleicher Tumorart unterscheiden sich Tumoren biologisch zwischen einzelnen Patienten sowie Patientengruppen. Das kann zu unterschiedlichen Behandlungserfolgen führen. Um die Patienten gezielter behandeln zu können und den Behandlungserfolg zu verbessern, ist es deshalb wichtig, die Therapie zu individualisieren. Dazu müssen die Tumorphenotypen charakterisiert werden, damit die Patienten klassifiziert werden können. Seit einigen Jahren liegt dabei der Forschungsschwerpunkt unter anderem auf der Nutzung von medizinischen Bilddaten. Die Verfahren der medizinischen Bildgebung (z. B. Computertomografie (CT) oder Magnet-Resonanz-Tomografie (MRT)) bieten den Vorteil, dass sie nichtinvasiv sind und dass die Bildgebung im klinischen Alltag schon vollkommen etabliert ist. Sie werden für die Bestrahlungsplanung sowie die Therapiekontrolle standardmäßig eingesetzt [1]. Um die Tumoren und damit auch die Patienten zu charakterisieren, werden quantitative Bildmerkmale (Features) aus den Bildern extrahiert. Basierend auf den Bildmerkmalen werden anschließend prognostische Modelle mittels maschineller Lernverfahren entwickelt, um das Patientenrisiko vorherzusagen. Dabei werden die Patienten entsprechend der Vorhersage in Hoch- und Niedrigrisikogruppen klassifiziert. Dieser Prozess wird auch „Radiomics“ genannt [29].

Ein klinisches Szenario ist, die Dosis der Strahlentherapie anhand der Risikoklassifizierung anzupassen. So könnten Patienten mit höherem Risiko eine höhere Dosis erhalten, um die Tumorheilungschancen zu erhöhen. Ebenso könnten Patienten mit niedrigerem Risiko eine verminderte Dosis appliziert bekommen, damit die Nebenwirkungswahrscheinlichkeit gesenkt wird.

In mehreren Studien konnten bereits prognostische Bildmerkmale und darauf basierende Risikomodelle für die Vorhersage des Therapieausgangs für verschiedene Tumorarten entwickelt

werden [1, 17, 20, 21, 22, 27]. Für gewöhnlich basieren Radiomics-Modelle auf mathematisch definierten Bildmerkmalen. Dabei ist Vorwissen erforderlich, wie die unterschiedlichen Tumorcharakteristiken beschrieben werden können [1, 20]. Es ist unklar, ob die so definierten Bildmerkmale den Tumor vollständig charakterisieren. Ein neuartiger Ansatz ist Deep-Learning, womit direkt die Bildcharakteristiken anhand der verwendeten Bildgebung erlernt werden können [8, 17, 22, 27, 29].
Um an diese Entwicklung anzuknüpfen, soll in dieser Arbeit die Risikovorhersage mittels Deep-Learning-Verfahren betrachtet werden. Es soll dabei untersucht werden, inwiefern sich Deep-Learning-Methoden eignen, um das Risiko von Kopf-Hals-Tumorpatienten nach der Strahlentherapie anhand ihrer CT-Bilder vorherzusagen. Die Genauigkeit soll mit konventionellen Machine-Learning-Algorithmen verglichen werden. Dabei wird als Therapieausgang das Gesamtüberleben (Overall Survival = OS) sowie die loko-regionäre Tumorkontrolle[1] (Loco-Regional tumor Control = LRC) untersucht.
Die Hypothese dieser Arbeit besteht darin, dass mithilfe der Deep-Learning-Methoden die Patienten besser klassifiziert werden können, als es mit den konventionellen Machine-Learning-Methoden der Fall ist.

[1] ob sich der Tumor nach der Strahlentherapie neu bildet bzw. ein loko-regionäres Rezidiv auftritt

2 Theoretischer Hintergrund

2.1 Röntgen-Computertomografie

Die röntgenbasierte Computertomografie stellt eines der wichtigsten bildgebenden Verfahren in der Medizin dar und wird zum Beispiel zur Diagnostik sowie zur Bestrahlungsplanung eingesetzt. Anhand der Schwächung von RÖNTGENstrahlen beim Durchgang durch einen Körper können Rückschlüsse auf dessen Zusammensetzung gezogen werden.

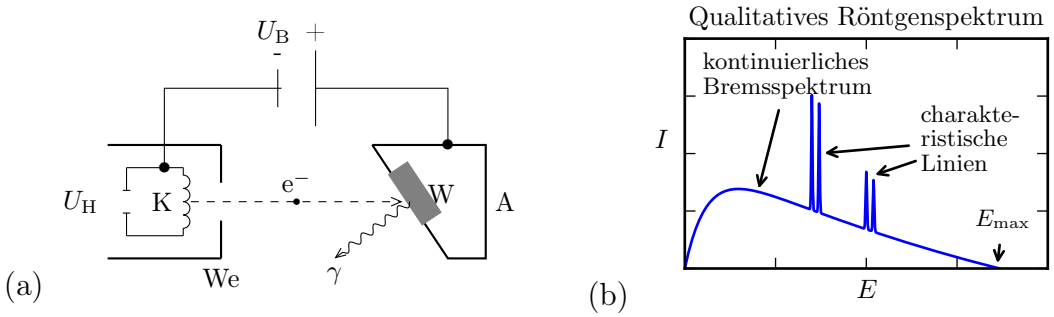


Abbildung 2.1: (a) Prinzipieller Aufbau einer RÖNTGENröhre [16]. A: Anode, K: Kathode, We: Wehneltzylinder, W: Wolframeinsatz. (b) Qualitatives RÖNTGENspektrum, in dem die Intensität I über der Energie E aufgetragen ist.

Die einfachsten RÖNTGENröhren zur Erzeugung von RÖNTGENstrahlung bestehen aus einer Glühkathode als Elektronenquelle und einer Wolframanode, zwischen denen eine Hochspannung im kV-Bereich anliegt (Abb. 2.1 (a)). Dadurch werden die von der Kathode emittierten Elektronen zur Anode hin durch ein Hochvakuum beschleunigt. Dort finden Wechselwirkungen mit den Elektronenhüllen der Targetatome statt, aus denen RÖNTGENstrahlung emittiert wird [16]. Durch Stoßionisation und darauf folgende Interbandübergänge entsteht monochromatische diskrete charakteristische RÖNTGENstrahlung. Im Spektrum wird diese durch kontinuierlich verteilte Bremsstrahlung überlagert (Abb. 2.1 (b)), welche durch die Abbremsung bzw. Umlenkung der Elektronen im atomaren COULOMBfeld entsteht. Jene kann maximal den Energiewert erreichen, der durch die maximale kinetische Energie der beschleunigten Elektronen definiert ist. Mit einer Beschleunigungsspannung U_B und unter der Annahme, dass die kinetische Energie der Elektronen direkt nach ihrer Emission vernachlässigbar ist, gilt also für die maximale Energie der Bremsstrahlung $E_{max} = E_{kin,e} = e \cdot U_B$ [16]. Für die medizinische Anwendung wird oftmals eine Beschleunigungsspannung zwischen 25 kV und 150 kV gewählt,

wobei für die Abbildung von kontrastarmen Gewebe niedrigere und für kontrastreicheres Gewebe höhere Spannungen genutzt werden [2, 16].

Bei der Schwächung der RÖNTGENstrahlung wird die Photonenflussdichte φ durch Wechselwirkung zwischen Photonen und Materie verringert [33]. Die wichtigsten beitragenden Effekte sind der Photoeffekt sowie die kohärente und inkohärente Streuung. Paarbildung und Kernphotoeffekt spielen bei den diagnostischen Photonenenergien keine Rolle, wogegen die inkohärente Streuung im Weichteilgewebe zwischen 60 keV und 1 MeV den dominierenden Prozess darstellt [12]. Beim Photoeffekt wird die gesamte Energie des einfallenden Photons an ein Hüllenelektron abgegeben, welches dadurch vom Atomrumpf gelöst wird (Photoelektron) [33]. Unter kohärenter Streuung versteht man die Streuung an gebundenen Elektronen der Atomhülle unter frequenz- und phasengleicher Abstrahlung, die mit einer Richtungsänderung einher geht. Die Schwächung eines schmalen Strahlenbündels entsteht dabei nur durch das „Herausstreuen“ aus dem Bündel [12]. Inkohärente Streuung ist dagegen immer mit einem Impuls- und Energieübertrag verbunden [12]. Zusammengefasst ergibt sich somit Gleichung (2.1) für die Photonenflussdichte nach dem Durchgang durch eine Materialschicht der Dicke x. Darin ist φ_0 die Flussdichte vor der Materialschicht, N_a die Teilchenzahldichte im Material und $\sigma_\mathrm{t,a}$ der totale atomare Wechselwirkungsquerschnitt. Das Produkt $\Sigma := N_\mathrm{a} \cdot \sigma_\mathrm{t,a}$ bezeichnet die Wechselwirkungsquerschnittsdichte und ist gleichzusetzen mit dem Schwächungskoeffizienten μ [33]. Dieser ergibt sich für zusammengesetzte Stoffe aus der Summe über die Einzelelemente [12];

$$\varphi(x) = \varphi_0 \cdot \exp(-N_\mathrm{a}\sigma_\mathrm{t,a} \cdot x) \ , \tag{2.1}$$

$$\mu = \sum_i N_{\mathrm{a},i} \cdot \sigma_{\mathrm{t,a},i} \ . \tag{2.2}$$

Gleichung (2.1) gilt dabei nur für schmale Strahlenbündel, da bei breiten Bündeln auch gestreute Strahlung im Detektor hinter der Materialschicht gemessen wird [33].

Mithilfe der Computertomografie lassen sich Schichtbilder eines kompakten, in der Regel inhomogenen Körpers erstellen. Dazu wird eine Schicht in ausreichend viele Volumenelemente (Voxel) unterteilt, denen jeweils ein Schwächungskoeffizient zugeordnet wird. Diese werden anschließend in einer (Bild-)Matrix angeordnet. Da die Schwächung hauptsächlich durch Wechselwirkungen mit den Hüllenelektronen entsteht, also abhängig von der Ordnungszahl Z der Targetatome ist, und die einzelnen Voxel unterschiedliche Dichten und Atomsorten beinhalten können, variieren die Schwächungskoeffizienten zwischen den Voxeln. Indem sie als analoge Grauwerte von Pixeln interpretiert werden, kann eine Schicht bildlich dargestellt werden. Die Grauwerte repräsentieren die Elektronendichten der jeweiligen Materialien, wie beispielsweise Luft, Wasser oder Weichteilgewebe (Organe und Bindegewebe). Die Auflösung wird durch die Größe der Bildmatrix bestimmt [25, 35]. Werden mehrere übereinander liegende Schichten aufgenommen, entstehen dreidimensionale (3D-) Bilder. Um die einzelnen Werte der Voxel zu erhalten, wird die Intensität der geschwächten Strahlung hinter dem Körper mit Detektoren ge-

messen. Aus dem Verhältnis der bekannten einfallenden und der austretenden Intensität lassen sich die über die Strahlrichtung addierten Schwächungskoeffizienten ermitteln. Solche transversalen Projektionen der zweidimensionalen (2D-) Verteilung der Schwächungskoeffizienten werden für mehrere polare Rotationswinkel (0° - 180°) aufgenommen und für die Berechnung der einzelnen Bildelemente verwendet. Die Rekonstruktion des Bildes geschieht dabei mithilfe der gefilterten Rückprojektion. Filter werden verwendet, um Artefakte[1] zu minimieren [25, 35].

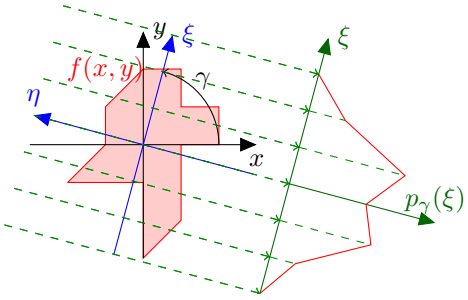


Abbildung 2.2: Das Prinzip der Projektion im mathematischen Sinne (Bild nach [36]). Im um γ gedrehten ξ-η-Koordinatensystem, wird die eindimensionale Projektion $p_\gamma(\xi)$ der zweidimensionalen Verteilung $f(x,y)$ durch Integration entlang der η-Achse berechnet.

Mathematisch kann die Bildrekonstruktion durch eine Darstellung der Bildmatrix als zweidimensionale Funktion $f(x, y)$ der Schwächungskoeffizienten in der x-y-Ebene beschrieben werden (außerhalb des Körpers gilt $f(x, y) = 0$). Dabei repräsentiert das x-y-Koordinatensystem den Körper und das ξ-η-Koordinatensystem die Position der Detektoren. Die eindimensionalen (1D-) Projektionen, welche in Realität durch die Schwächung der in einer Richtung verlaufenden RÖNTGENstrahlen entstehen, berechnen sich als Integration entlang der η-Achse (Abb. 2.2) in dem um γ rotierten ξ-η-Koordinatensystem [2]

$$\begin{bmatrix} \xi \\ \eta \end{bmatrix} = \begin{bmatrix} \cos\gamma & \sin\gamma \\ -\sin\gamma & \cos\gamma \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} , \tag{2.3}$$

$$p_\gamma(\xi) = \int_{-\infty}^{\infty} f(\xi\cos\gamma - \eta\sin\gamma, \xi\sin\gamma + \eta\cos\gamma)\,\mathrm{d}\eta \ . \tag{2.4}$$

Zur Erläuterung des Grundprinzips wird im Folgenden lediglich die tomografische Berechnung der Translations-Rotations-CTs beschrieben, bei denen die Werte für eine Projektion entlang paralleler Linien aufgenommen werden. Moderne CT-Geräte der dritten und vierten Generation verwenden hingegen einen aufgefächerten RÖNTGENstrahl, um die Translationszeit der RÖNTGENquelle zu reduzieren (Abb. A.1). Die Umwandlung der 2D-Verteilung $f(x,y)$ in die 1D-Projektionen $p_\gamma(\xi)$ für mehrere Winkel γ ($\gamma \in [0°, 180°]$) entspricht einer RADONtransformation. Anschließend wird das FOURIER-Scheiben-Theorem verwendet, das die Identität eines linearen radialen Schnittes der 2D-FOURIERtransformierten $\tilde{F}(u,v) =$

[1] Bildstörungen, die aufgrund der Berechnung auftreten

$\mathcal{F}_2(f(x,y))$ unter dem Winkel γ mit der 1D-FOURIERtransformierten $\tilde{P}_\gamma(q) = \mathcal{F}_1(p_\gamma(\xi))$ der gemessenen RADONtransformation $p_\gamma(\xi)$ beschreibt (Herleitung in Kap. A.2) [2]

$$\begin{aligned} \begin{bmatrix} u \\ v \end{bmatrix} &= q \cdot \begin{bmatrix} \cos\gamma \\ \sin\gamma \end{bmatrix} \qquad \text{(Koordinatentransf. im FOURIER-Raum)} , \quad (2.5) \\ \mathcal{F}_2(f(x,y)) &= \tilde{F}(u,v) = \tilde{F}(u(q,\gamma), v(q,\gamma)) , \\ \tilde{F}(u(q,\gamma), v(q,\gamma)) &= \tilde{F}(q,\gamma) = \tilde{P}(q,\gamma) \equiv \tilde{P}_\gamma(q) . \quad (2.6) \end{aligned}$$

Bei einer CT-Aufnahme werden mit Detektoren die Projektionen $p_\gamma(\xi)$ der 2D-Verteilung der Schwächungskoeffizienten gemessen, die FOURIERtransformierten $\tilde{P}_\gamma(q)$ davon berechnet und im Frequenzraum eine Rücktransformation von Polar- in kartesische Koordinaten durchgeführt (Glg. (2.5)). Damit wird die FOURIERtransformierte $\tilde{F}(u,v)$ von $f(x,y)$ konstruiert und anschließend die inverse FOURIERtransformation angewendet, um die ursprüngliche Verteilung $f(x,y)$ zu erhalten [2]. Durch die endliche Anzahl von gemessenen Projektionen sind lediglich endlich viele radiale Linien (im FOURIER-Raum) von $\tilde{F}(u,v)$ bekannt, sodass eine formale Rücktransformation durch Integration nicht einfach möglich ist. Aus den fehlenden Informationen resultieren Abbildungsfehler (Artefakte), die das Bild zum Teil verfälschen. Durch Faltung der Projektionen mit einer Filterfunktion können die Abbildungsfehler zum Teil unterdrückt bzw. korrigiert werden. Die Anzahl der Schwächungsmessungen aus unterschiedlichen Richtungen bestimmt jedoch auch die Anzahl und den Informationsgehalt der einzelnen Bildelemente und damit die räumliche Auflösung [2, 35, 36].
Angegeben werden die resultierenden Werte in der Bildmatrix in HOUNSFIELD-Einheiten (HEs). Die HOUNSFIELD-Skala setzt die mittleren Schwächungskoeffizienten der einzelnen Voxel $\tilde{\mu}$ ins relative Verhältnis zum Schwächungskoeffizient von Wasser μ_{H_2O} bei monochromatischer Strahlung von 73 keV,

$$\text{CT-Zahl} = \frac{\tilde{\mu} - \mu_{H_2O}}{\mu_{H_2O}} \cdot 1000\,\text{HE} . \quad (2.7)$$

Damit ergibt sich für Wasser die CT-Zahl von 0 HE und für Luft -1000 HE. Knochen erreichen CT-Zahlen von 1000 HE und mehr. Da somit mehr als 2000 verschiedene Grauwerte entstehen, das menschliche Auge aber nur ca. 100 unterscheiden kann, wählt man zur CT-Darstellung nur ein gewisses HE-Fenster aus. Dadurch wird die Grauwertskala gespreizt und es können mehr Strukturen im CT-Bild erkannt werden.

2.2 Deep-Learning und Neuronale Netze

Deep-Learning ist eine Methode des maschinellen Lernens[2], die sich in den letzten Jahren als sehr effektiv erwiesen hat, um repräsentative Strukturen in hochdimensionalen Daten zu

[2] engl. machine learning

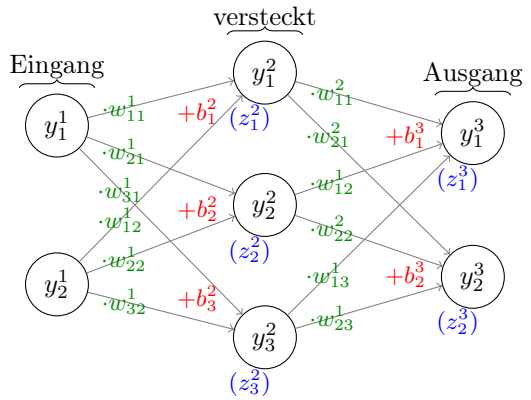
erlernen [19]. Deshalb wird es mittlerweile in vielen Aufgabenfeldern, wie zum Beispiel der Bild- und Spracherkennung oder auch zur Datenanalyse von Teilchenbeschleunigern, eingesetzt [19]. Es werden anhand einer Sequenz aus Operationen (z. B. Faltung) abstrakte Datenrepräsentationen erlernt, welche (je nach Aufgabe) zur Klassifikation der Daten genutzt werden können [19]. In vielen Klassifikationsaufgaben soll die Eingang-Ausgang-Funktion des Modells unsensitiv gegenüber irrelevanten Variationen in den Daten sein (bei der Bilderkennung z. B. Position, Orientierung, Belichtung des Bildobjekts). Dagegen soll sie besonders sensitiv gegenüber Variationen in den Strukturen der Daten sein, welche für die Klassifizierungsaufgabe von Bedeutung sind [19]. Dabei können die relevanten Variationen (z. B. Unterschied zwischen Schäferhund und Wolf auf Bildern) verglichen zu den irrelevanten (z. B. Pose des Tiers oder Position im Bild) viel kleiner erscheinen [19]. Konventionelle Klassifikatoren können nur schwer zwischen relevanten und irrelevanten Variationen unterscheiden. Deshalb wird bei konventionellen Machine-Learning-Techniken ein gesonderter Merkmal-Extrahierer benötigt, der die einzelnen Rohdaten bezüglich der Klassifikationsaufgabe in passende Repräsentationen (Merkmal-Vektoren) umformt [19]. Erst danach kann der Klassifikator mittels maschinellen Lernens anhand der Repräsentationen und ihren Klassenzuordnungen traininert werden. Einen Merkmals-Extrahierer zu entwickeln, der gute Repräsentationen für das Training liefert, benötigt viel Aufwand und Expertise [19]. Das kann beim Deep-Learning umgangen werden, da das Modell die wichtigen Merkmale aus den Daten automatisch erlernt. Das Schlüsselkonzept ist hierbei, dass die Merkmale nicht explizit mathematisch definiert werden müssen, sondern durch allgemeine Lernprozeduren ermittelt werden. Es werden Rückwärtspropagierungs-Algorithmen verwendet, um herauszufinden, wie die inneren Parameter des mehrschichtigen Rechenmodells angepasst werden müssen. Mithilfe dieser Parameter werden die Repräsentationen in einer Schicht des Modells aus der vorangehenden Schicht berechnet [19]. Als mehrschichtige Rechenmodelle dienen meistens künstliche neuronale Netze[3] (NNs).

2.2.1 Neuronale Netze

Neuronale Netze bestehen aus mehreren Schichten, die aus Neuronen aufgebaut sind [28]. Sie bilden einen Eingang mit fester Größe auf einen Ausgang mit fester Größe ab, indem sie ihn durch einfache Rechenoperationen durch die Schichten propagieren. Dabei werden die Neuronen einer Schicht durch die Neuronen der vorangegangenen Schicht beeinflusst. Dieses Prinzip greift die Idee auf, dass viele natürliche Signale aus Hierarchien bestehen, was bedeutet, dass sich Merkmale höherer Stufen aus solchen niedrigerer Stufen zusammensetzen [19]. Die einfachsten Netze bestehen aus vollvernetzten (fc-) Schichten[4], in denen die Werte der Neuronen durch eine gewichtete Summe der Neuronenwerte der vorangegangenen Schicht und der anschließenden Anwendung einer oft nichtlinearen Aktivierungsfunktion f berechnet werden

[3] im folgenden nur noch als neuronale Netze bezeichnet
[4] engl. fully connected (fc) layers

(Abb. 2.3 und Glgn. (2.8), (2.9)) [19, 28]. Somit ist jedes Neuron einer fc-Schicht durch reelle Zahlen (Gewichte) mit allen Neuronen der vorangehenden bzw. folgenden Schicht verbunden. Diese Gewichte definieren in ihrer Gesamtheit die Eingang-Ausgang-Funktion des Netzwerks und bestehen in praktischen Anwendungen häufig aus mehr als einer Million Zahlen. Als Aktivierungsfunktion kann theoretisch jede stetige Funktion dienen, jedoch wird heutzutage häufig die *ReLU*-Funktion[5] ($ReLU(x) = \max(0, x)$) verwendet, da sich gezeigt hat, dass Netzwerke mit vielen Schichten dann schneller lernen [19]. Schichten im Innern des Netzwerks, die weder Eingang noch Ausgang darstellen, werden versteckte Schichten[6] genannt. Durch die Rechenschritte im Inneren des (aus mehreren Neuronen bestehenden) Netzwerks wird der Eingang nichtlinear „verzerrt“, sodass beim Ausgang die Kategorien, in die die Daten eingeteilt werden sollen, linear seperabel werden [19].



$$\vec{z}^{\,l} = W^{l-1} \cdot \vec{y}^{\,l-1} + \vec{b}^{\,l} \tag{2.8}$$

$$\vec{y}^{\,l} = f(\vec{z}^{\,l}) \tag{2.9}$$

Beispiel:

$$\begin{bmatrix} z_1^2 \\ z_2^2 \\ z_3^2 \end{bmatrix} = \begin{bmatrix} w_{11}^1 & w_{12}^1 \\ w_{21}^1 & w_{22}^1 \\ w_{31}^1 & w_{32}^1 \end{bmatrix} \cdot \begin{bmatrix} y_1^1 \\ y_2^1 \end{bmatrix} + \begin{bmatrix} b_1^2 \\ b_2^2 \\ b_3^2 \end{bmatrix}$$

$$\begin{bmatrix} y_1^2 \\ y_2^2 \\ y_3^2 \end{bmatrix} = f\left(\begin{bmatrix} z_1^2 \\ z_2^2 \\ z_3^2 \end{bmatrix} \right) = \begin{bmatrix} f(z_1^2) \\ f(z_2^2) \\ f(z_3^2) \end{bmatrix}$$

Abbildung 2.3: Funktionsweise eines neuronalen Netzes mit vollvernetzten Schichten. Die Berechnung der Neuronenwerte y_i^l, $i \in \{1, \ldots, n\}$ der Schicht l mit n Neuronen erfolgt durch die Gleichungen (2.8) und (2.9). Die gewichtete Summe kann kompakt als Matrixgleichung aufgeschrieben werden. z_i^l bezeichnet den Aktivierungswert des i-ten Neurons in Schicht l. W ist die Gewichtematrix, in der w_{ij}^{l-1} das Gewicht ist, dass das j-te Neuron aus Schicht $l-1$ mit dem i-ten Neuron in Schicht l verbindet. Jedem Neuron ist zusätzlich noch ein Strafterm (Bias) b zugeordnet, der zur gewichteten Summe hinzu addiert wird. Um den Ausgang des Neurons zu erhalten, wird die Aktivierungsfunktion f auf den Aktivierungswert angewendet (auf Vektoren komponentenweise) [19, 28].

Eine wichtige Weiterentwicklung der NNs sind faltende Netze[7] (CNNs), die besonders gut mehrdimensionale Daten-Arrays (z. B. RGB-Bilder) verarbeiten können [19]. Seit Beginn der 2000er Jahre wurden mit CNNs große Erfolge in der maschinellen Auffindung, Segmentierung und Zuordnung von Objekten oder Regionen in Bildern erreicht [19]. Die Besonderheit der CNNs sind die Faltungs- und Auswahlschichten[8] (Conv- und Pool-Schichten) anstelle der fc-Schichten zu Beginn der Netze (Abb. 2.4) [19]. Die Neuronen der Conv-Schichten sind in Merkmalsabbildern angeordnet und jeweils mit einem lokalen Bereich eines Merkmalsabbildes

[5] ReLU: Rectified Linear Unit
[6] engl. hidden layers
[7] engl. Convolutional Neuronal Networks
[8] engl. convolutional & pooling layers

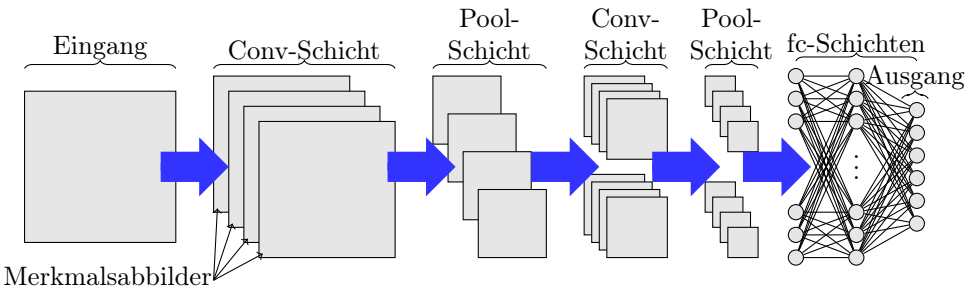


Abbildung 2.4: Netzwerkarchitektur eines faltenden Netzwerks. Beispiel mit jeweils zwei Faltungs- und Auswahlschichten (Conv- und Pool-Schichten) als Basis vor den vollvernetzten Schichten (mit einer Klassifikation in sechs Klassen).

aus der vorherigen Schicht verbunden. Die Gewichte zur Berechnung der gewichteten Summe über einen solchen Bereich bilden einen Filter und innerhalb eines Merkmalsabbildes wird für die Berechnung der Neuronenwerte immer der gleiche Filter benutzt (Prinzip in Abb. A.2). Verschiedene Merkmalsabbilder einer Conv-Schicht werden dagegen durch unterschiedliche Filter berechnet. Die Idee hinter diesem Prinzip ist, dass in Datenarrays oft lokale Gruppen von Werten hochgradig korreliert und lokale Statistiken invariant gegenüber ihrer Lokalisation sind. So kann zum Beispiel ein Bildmotiv an verschiedenen Stellen des Bildes auftreten. Deshalb werden den Neuronen an verschiedenen Stellen gleiche Gewichte zugeordnet, damit sie die gleichen Strukturen nachweisen können [19]. Die Pool-Schichten verbinden bedeutungsmäßig gleiche Bildmerkmale zu einem Bildmerkmal, indem sie beispielsweise das Maximum über einen lokalen Bereich eines Merkmalsabbildes bilden (Maxpooling). Dadurch reduzieren sie die Dimension der Repräsentation und stellen ihre Invarianz gegenüber kleinen Verschiebungen oder Verzerrungen her (Prinzip in Abb. A.3). Die Rückwärtspropagierungs-Algorithmen (Kap. 2.2.2) funktionieren bei Conv-Schichten analog zu fc-Schichten [19].

2.2.2 Deep-Learning-Prozess und Rückwärtspropagierung

Die übliche Methode, um CNN-Netze zu trainieren, ist das „beaufsichtigte Lernen“[9]. Dazu wird ein Datensatz verwendet, bei dem jedem Datenpunkt eine Klassenzugehörigkeit (Label) zugeordnet ist. Durch die Vielzahl der Netzwerkparameter sollte optimaler Weise der Umfang des Datensatzes die Anzahl der Netzwerkparameter übersteigen. Während des Trainings werden die Einzeldaten als Eingang in das Netz gegeben, welches mit Rechenpropagation durch das Netz einen Ergebniswert zu jeder möglichen Klasse als Ausgang produziert. Jeder Datenpunkt wird durch das Netzwerk in die Klasse mit dem höchsten Ergebniswert eingeordnet. Die vorhergesagte Klasse stimmt in der Regel nicht sofort mit der richtigen Klasse überein, was den Ausgangspunkt für die Rückwärtspropagierung - den Lernalgorithmus - bildet. Eine Zielfunktion (Loss-Funktion) misst die Abweichung zwischen Ausgang und gewünschtem Ergebnis und mittelt die Abweichung über alle Trainingsbeispiele. Die Parameter des Netzwerks (Gewichte)

[9] engl. supervised learning

werden so angepasst, dass die mittlere Abweichung minimiert wird. Dazu wird der negative Gradient der Loss-Funktion mithilfe der Rückwärtspropagierung bestimmt [19]. Diese ist eine einfache Anwendung der Kettenregel. Um die Ableitung der Loss-Funktion bezogen auf den Eingang eines Neurons zu berechnen, wird sich wegen des schichtweisen Aufbaus des Netztes rückwärts durch das Netzwerk und somit rückwärts durch den Gradienten gearbeitet, um ihn zu bestimmen [19, 28].

Als Beispiel dient das Netzwerk, das in Abbildung 2.3 dargestellt ist. Der Zielwert sei $\vec{t}$, der Vorhersagewert $\vec{y}^3$ und die Aktivierungsfunktion f. Die Loss-Funktion sei durch $C(\vec{y}^3, \vec{t}) := 1/2 \cdot (\vec{y}^3 - \vec{t})^2$ definiert. Ausgehend von der Ausgangs-Schicht werden die partiellen Ableitungen der Loss-Funktion nach den einzelnen Gewichten und Straftermen (Bias) bestimmt. So ergibt sich zum Beispiel für alle Gewichte zwischen der zweiten und dritten Schicht:

$$\frac{\partial C}{\partial y_1^3} = y_1^3 - t_1 \ , \tag{2.10}$$

$$\frac{\partial C}{\partial b_1^3} = \frac{\partial C}{\partial z_1^3}\frac{\partial z_1^3}{\partial b_1^3} = \frac{\partial C}{\partial y_1^3}\frac{\partial y_1^3}{\partial z_1^3}\frac{\partial z_1^3}{\partial b_1^3} = (y_1^3 - t_1) \cdot f'(z_1^3) \cdot 1 \ , \tag{2.11}$$

$$\vdots$$

$$\frac{\partial C}{\partial w_{23}^2} = \frac{\partial C}{\partial z_2^3}\frac{\partial z_2^3}{\partial w_{23}^2} = \frac{\partial C}{\partial y_2^3}\frac{\partial y_2^3}{\partial z_2^3}\frac{\partial z_2^3}{\partial w_{23}^2} = (y_2^3 - t_2) \cdot f'(z_2^3) \cdot y_3^2 \ . \tag{2.12}$$

Nach der Berechnung der Ausdrücke wird zur nächsten Schicht übergegangen, indem weiter zurück propagiert wird:

$$\frac{\partial C}{\partial y_1^2} = \sum_{i=1}^{2} w_{i1}^2 \cdot \frac{\partial C}{\partial z_i^3} \ , \tag{2.13}$$

$$\vdots \quad .$$

Schicht für Schicht werden alle partiellen Ableitungen gebildet. Diese ergeben zusammen den gesuchten Gradienten, in dessen negative Richtung die Gewichte angepasst werden [19, 28]. In der Praxis werden im Training stochastische Methoden wie Stochastic Gradient Descent (SGD) verwendet, um Arbeitsspeicher und Rechenzeit zu sparen. Pro Trainingsschritt wird dem Netzwerk nur eine zufällig ausgewählte Teilmenge des gesamten Trainingssatzes gezeigt. Damit wird eine Abschätzung des über alle Trainingsdaten gemittelten Gradienten zur Parameteranpassung erhalten, bevor der nächste Teildatensatz in das Netzwerk gegeben wird. Nach dem Training wird die Leistung des Netzwerks an einem Testdatensatz überprüft, um die Verallgemeinerungsfähigkeit[10] des Netzwerks zu testen [19].

Durch die schrittweise Anpassung der Parameter im Training kann die Optimierung bei einem lokalen Minimum der Loss-Funktion zum Erliegen kommen, ohne dass das globale Minimum durch den Algorithmus erreicht wird. Daher werden die Gewichte möglichst passend durch

[10] die Fähigkeit, Vorhersagen auf neue, im Training nie gesehene Eingangsdaten, zu liefern

Vortraining mit anschließender Feinjustierung initialisiert. Diese Methoden werden vor allem bei kleinen Datensätzen eingesetzt [19]. Ein weiteres Problem sind Überanpassungseffekte: Das Netzwerk passt dabei die Parameter bezogen auf die Trainingsdaten sehr gut an, verliert jedoch seine Verallgemeinerungsfähigkeit. Zur Vermeidung von Überanpassungseffekten werden Dropout und Lernregulierungen in das Netzwerk integriert. Dropout bezeichnet das Ignorieren eines gewissen Anteils der Gewichte bei der Parameteranpassung während eines Trainingsschrittes. Die Lernregulierung bezieht sich auf die Lerndauer (Anzahl der Epochen[11]) oder die Lernrate, welche die Größe der Schritte für die Parameteranpassung definiert [28, 32].

2.3 Radiomics

Unter Radiomics versteht man den Prozess der Extraktion und Analyse großer Mengen an quantitativen Bildmerkmalen von medizinischen Bilddaten (CT-, MRT-, Positronen-Emissions-Tomografie-Bilder) [29]. Dazu gehört die umfassende Quantifizierung und Charakterisierung von Tumorphenotypen, wozu oftmals maschinelles Lernen verwendet wird [1, 20]. Die Bildmerkmale werden benutzt, um prognostische Modelle zu entwickeln. Unter Verwendung von molekularen Daten können auch Verbindungen zu den zugrunde liegenden biologischen Eigenschaften der Tumoren (Radiogenomics) hergestellt werden [1, 17].
Im Folgenden sollen die Radiomics-Ansätze, welche auf konventionellen Machine-Learning Algorithmen basieren, von denen, die auf Deep-Learning basiert sind, unterschieden werden. Bei der konventionellen Methode werden die zu extrahierenden Bildmerkmale explizit mathematisch durch den Anwender definiert. Diese Bildmerkmale quantifizieren zum Beispiel die Intensität, die Tumorform und die Tumorzusammensetzung im Bild oder sind Wavelet-basiert. Um die große Anzahl dieser Bildmerkmale zu reduzieren, werden stark korrelierte Bildmerkmale zusammengefasst. Anschließend wird eine Bildmerkmal-Selektion durchgeführt, wobei die Merkmale auf Robustheit[12], Stabilität und Korrelation mit dem betrachteten Endpunkt geprüft werden, sodass redundante und nicht mit dem Therapieausgang korrelierte Merkmale vernachlässigt werden können. Die übrig gebliebenen, aussagekräftigen Bildmerkmale bilden eine Radiomics-Signatur, anhand der mithilfe von maschinellem Lernen ein Radiomics-Modell erstellt wird. Für die genannten Schritte werden Trainingsdaten verwendet. Anschließend wird die Vorhersagekraft der Signatur und des Modells mit Testdaten validiert (Abb. 2.5) [1, 20]. Das Mengenverhältnis zwischen Trainings- und Testdatensatz wird oftmals $2:1$ gewählt [20].
Für Deep-Learning-Radiomics gibt es grundsätzlich zwei Varianten. In der ersten Variante wird ein NN trainiert, um die Klassifikation von Anfang bis Ende durchzuführen. Dabei wird die Klassifikation anhand eines „rohen“ Bildes direkt vom NN vorgenommen. Die andere Variante ersetzt lediglich die konventionell mathematisch definierten Bildmerkmale durch Bildmerkma-

[11] Trainingsabschnitt, innerhalb dessen alle Trainingsdaten einmal in das Netzwerk gegeben wurden
[12] Fähigkeit der Bildmerkmale z.B. auch bei eventuellen Bildstörungen den selben Wert zu liefern

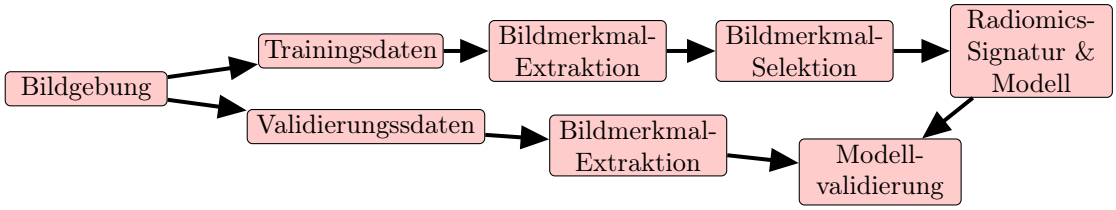


Abbildung 2.5: Ablaufschema zur Entwicklung eines Radiomics-Modells.

le, die mit Deep-Learning-Algorithmen geniert wurden (Deep-Bildmerkmale). Diese Bildmerkmale bestehen aus Neuronenwerten versteckter Schichten und werden nach der Extraktion auf konventionelle Weise, wie oben beschrieben, weiterverarbeitet (→ Bildmerkmal-Selektion → Klassifizierung). Die zugrunde liegende Idee ist, dass selbst definierte Bildmerkmale oft von niedriger Ordnung sind (flache Bildmerkmale). Zum Beispiel wird vermutet, dass die intra-Tumor-Heterogenität in den Bildern mit der genetischen Heterogenität verbunden ist [17]. Diese steht wiederum mit der Aggressivität des Tumors in Verbindung. Selbstdefinierte flache Bildmerkmale könnten diese Heterogenität nicht komplett charakterisieren, wodurch das Potential der Radiomics-Modelle begrenzt sein kann. Deep-Bildmerkmale hingegen sind von höherer Ordnung und Abstraktion. Damit könnten Tumorphenotypen vielleicht besser charakterisiert werden bzw. prognostisch relevantere Strukturen gefunden werden, was die Leistungsfähigkeit der Modelle verbessern könnte [17, 29]. Für solche Aufgaben werden meistens CNNs verwendet (CNN-Bildmerkmale), da diese bereits gute Klassifizierungsleistungen in anderen Domänen des maschinellen Bildverstehens gezeigt haben [19]. Da medizinische Datensätze oft nicht groß genug sind, um ein NN von Grund auf trainieren zu können, wird oft auf Lerntransfer[13] zurückgegriffen [17, 29]. Dabei werden Netze verwendet, die schon für andere Klassifizierungsaufgaben in anderen Domänen trainiert wurden. Bestimmte erlernte Strukturen, die für die ursprüngliche Klassifikation von Bedeutung waren, können dabei auf die neue Klassifizierungsaufgabe übertragen werden [7, 9, 17, 29, 37].
Übergeordnetes Ziel ist das Entschlüsseln eines allgemeinen prognostischen Phenotyps, der unter Umständen in mehreren Krebsarten existiert [1].

[13] engl. Transfer-Learning

3 Material und Methoden

Im Rahmen dieser Arbeit wurden Patientendaten aus verschiedenen deutschen Studienzentren für die Untersuchungen verwendet. Ein Patientendatensatz bestand aus prätherapeutischen CT-Bildern, Tumormasken[1], welche von Radioonkologen in den jeweiligen Zentren erstellt wurden (beides in DICOM-Format [26]) und Therapieergebnissen[2]. Die DICOM-Dateien eines CT-Bildes und einer Tumormaske wurden jeweils in eine NIfTI-Datei [18] umgewandelt, sodass für jeden Patienten ein 3D-CT-Bild und eine 3D-Tumormaske zur Verfügung stand. Die gesamte Patientenkohorte wurde in einen Trainings- und einen Validierungsdatensatz (Kap. 3.1) unterteilt. Die Therapieergebnisse wurden binarisiert (Kap. 3.2), um die Patienten in zwei Gruppen (Hoch- und Niedrigrisiko) zu klassifizieren. Schließlich wurden, falls nötig, mithilfe der 3D-CTs die Tumormaskierungen korrigiert und aus diesen 2D-Schichtbilder extrahiert (Kap. 3.4). Die Bilder wurden vorprozessiert (Kap. 3.4), um den Anforderungen der gewählten Netzarchitektur[3] (Kap. 3.3) zu genügen. Aus den fertigen Teilbildern[4] wurden verschiedene Datensätze konstruiert, mit denen das Netzwerk mittels Deep-Learning trainiert wurde (Kap. 3.5). Aus speziell ausgewählten Datensätzen wurden Bildmerkmale mit dem trainierten Netz extrahiert und die Vorhersagekraft der Bildmerkmale untersucht (Kap. 3.6).

3.1 Patienten-Kohorten

Für die Untersuchungen standen die Daten von insgesamt 302 Patienten mit fortgeschrittenen Kopf-Hals-Tumoren zur Verfügung, die mit einer primären Radiochemotherapie behandelt wurden. Dazu wurden Patienten aus bestehenden Kohorten verschiedener Studien zusammengefasst. Für jeden Patienten waren ein CT-Bild, eine Tumormaske (auch ROI[5]-Maske) und die patientenspezifischen klinischen Daten zum Therapieergebnis vorhanden. Das Therapieergebnis wird durch den Ereignisstatus sowie eine dazugehörige Ereigniszeit charakterisiert. Der Ereignisstatus gibt an, ob das betreffende Ereignis (z. B. loko-regionäres Rezidiv, LRC oder Tod des Patienten, OS) eingetreten ist (1) oder nicht (0). Für die Individualisierung der

[1] Lokalisieren den Tumor im vorhandenen CT-Bild. Sie bestehen ebenfalls aus Voxeln (exakt so viele wie im CT-Bild), die jedoch nur mit Nullen und Einsen anstatt mit Grauwerten belegt sind. Alle Voxel mit dem Eintrag 1 gehören zum Tumor, während die 0-Voxel gesundes Gewebe oder Umgebung repräsentieren.
[2] z. B. wann ein loko-regionäres Rezidiv aufgetreten oder wann ein Patient verstorben ist
[3] Aufbau eines neuronalen Netzes
[4] Vorprozessierte 2D-Schichtbilder
[5] engl. Region of Interest

Strahlentherapie ist LRC das interessantere Ereignis, da bei den verstorbenen Patienten nicht eindeutig ist, ob der Tod durch die eigentliche Krebserkrankung, die Therapie oder eine andere Ursache hervorgerufen wurde. Bei einem loko-regionärem Rezidiv ist jedoch davon auszugehen, dass die Krebszellen nicht vollständig abgetötet wurden. Die Ereigniszeit gibt die Zeitspanne zwischen Therapiebeginn eines Patienten und dem Auftreten des jeweiligen Ereignisses bzw. dem Ende der Studiendauer (wenn das Ereignis nicht eingetreten ist) an.
Für die Einteilung der Patienten in eine Trainings- und eine Validierungskohorte wurde sich an früheren Arbeiten der Arbeitsgruppe orientiert, um intern eine bessere Vergleichbarkeit der Ergebnisse zu erhalten (z.B. [20]). Die Trainingskohorte bestand aus 207 DKTK[6]-Patienten [23]. Die restlichen 95 Patienten, bestehend aus 51 Studienpatienten des UKD[7] [24, 38] und 44 Patienten aus anderen Studien, dienten als Validierungskohorte.

3.2 Binarisierung des Therapieausgangs

Um ein neuronales Netz so zu trainieren, dass es Patienten anhand des CT-Datensatzes in eine Hoch- und Niedrigrisikogruppe einteilen kann, müssen die Patienten der Trainingskohorte zuvor den entsprechenden Gruppen zugeordnet werden. Dieser Prozess wird als Binarisierung der Therapieergebnisse bezeichnet, da die zwei Gruppen durch Null und Eins repräsentiert werden. Dabei wurden die kontinuierlichen Ereigniszeiten so unterteilt, dass zusammen mit dem Ereignisstatus zwei diskrete Klassen bzw. Gruppen von Patienten definiert wurden. Die Patientendaten wurden jeweils für beide Endpunkte LRC und OS getrennt binarisiert. Die Stichzeit für die Einteilung in Hoch- und Niedrigrisikogruppe wurde auf 24 Monate nach dem Therapiebeginn festgelegt. Das heißt, alle Patienten, bei denen das Ereignis innerhalb der ersten 24 Monate nach dem Behandlungsbeginn eingetreten war, wurden der Hochrisikogruppe (repräsentiert durch Null) zugeordnet und alle Patienten, bei denen das Ereignis später oder gar nicht eingetreten war, der Niedrigrisikogruppe (repäsentiert durch Eins). Es gab Patienten, die vor dem Ablauf der 24 Monate nicht mehr zu den Nachsorgeuntersuchungen erschienen, jedoch ereignisfrei waren. Bei diesen Patienten ist nicht klar, ob sie auch 24 Monate lang ohne Ereignis waren, weil dafür die nötigen Informationen fehlen. Um eine zufällig falsche Einordnung dieser Patienten bezüglich des jeweiligen Ereignisses zu vermeiden, wurden diese aus den entsprechenden Kohorten ausgeschlossen (Flussdiagramm der gesamten Binarisierung in Abb. A.4). Dadurch wurden 73 der 302 Patienten für die Betrachtung von LRC und 23 für die Betrachtung von OS ausgeschlossen. Die Patientenzahlen der Kohorten und ihre Gruppenzusammensetzungen sind in Tabelle 3.1 zusammengefasst. Der relative Anteil von Hoch- (0) und Niedrigrisikogruppe (1) zwischen der jeweiligen Trainings- und Validierungskohorte ist annähernd gleich groß. Dadurch ist zu vermuten, dass keine künstlichen Tendenzen er-

[6]Deutsches Konsortium für Translationale Krebsforschung
[7]Universitätsklinikum Carl Gustav Carus Dresden

Tabelle 3.1: Gruppenzusammensetzung und Patientenzahlen der Kohorten. Zahlen in Klammern geben den relativen Anteil an der Gesamtzahl der Kohorte an.

Kohorte	Patientenzahl gesamt	Patientenzahl Gr. 0	Patienzahl Gr. 1
LRC-Komplett	229	105 (45,9 %)	124 (54,1 %)
LRC-Training	158	75 (47,5 %)	83 (52,5 %)
LRC-Validierung	71	30 (42,3 %)	41 (57,7 %)
OS-Komplett	279	137 (49,1 %)	142 (50,9 %)
OS-Training	188	94 (50,0 %)	94 (50,0 %)
OS-Validierung	91	43 (47,3 %)	48 (52,7 %)

zeugt werden. Da die Anteile jeweils zwischen 40 % und 60 % liegen, ist ein ausgeglichenes Netztraining möglich.

3.3 Wahl der Netzarchitektur

Das zu trainierende Netzwerk wurde so wie alle anderen Programme für die folgenden Arbeitsschritte selbst mit Python (Version 2.7) implementiert. Die genutzten Deep-Learning-Grundstrukturen wurden dabei durch Keras [3] mit Theano [34] als Unterbau bereitgestellt. Diese beinhalteten die grundlegenden Bausteine und Funktionen für den Aufbau und das Training eines neuronalen Netzes. Es wurde ein CNN verwendet, da diese Sorte von NNs in der Vergangenheit die besten Ergebnisse in der Klassifikation von Bildern gezeigt hat (vgl. [19]). Um die mehreren Millionen Parameter eines NNs von Grund auf zu trainieren, reichen 158 bzw. 188 Trainingsdaten nicht aus, sodass Lerntransfer genutzt wurde. Dabei wird angenommen, dass ein auf anderen Bildern trainiertes neuronales Netz grundlegende Bildstrukturen erlernt hat, mithilfe deren das Netz auch anhand neuen Bildern Entscheidungen treffen kann [7, 9, 29, 37]. Diese Methode wurde auch in mehreren vorangehenden Studien verwendet und zeigte darin gute Ergebnisse (z.B. [17, 29]).
Der verwendete Netzaufbau wurde aus einer Vielzahl existierender Netzarchitekturen auf Grundlage bestehender Literatur zu Radiomics-Studien, in denen Deep-Learning eingesetzt wurde, ausgewählt [17, 22, 27, 29]. In diesen Studien wurden relativ kleine bzw. „flache“ Netze mit wenigen Schichten genutzt. Oftmals bestanden diese Netze aus vier bis sechs Conv-Schichten, gefolgt von drei fc-Schichten, wobei die dritte fc-Schicht den Netzausgang darstellte. Die Netzauswahl war auf Grund des Lerntransfers auf vortrainierte Netze beschränkt, die von Keras bereit gestellt werden[8]. Diese sind alle um einige Schichten tiefer (23 bis 572 Schichten insgesamt), als die in der Literatur verwendeten Architekturen. Da in anderen Studien bereits ähnliche Architekturen (auch VGG-Architekturen) verwendet wurden, wurde das Netz VGG16 gewählt [29, 30]. Dieses ist mit den insgesamt 23 Schichten das kleinste für Keras be-

[8] `https://www.keras.io/applications/`

reitgestellte vortrainierte CNN. Der grundlegende Aufbau ähnelt den Netzen in [29]. Anstelle der fünf Conv-Schichten werden fünf Conv-Blöcke verwendet. Die ersten beiden Conv-Blöcke bestehen aus zwei, die letzten drei Conv-Blöcke aus drei Conv-Schichten. Nach jedem Block ist eine Maximums-Auswahlschicht eingefügt, um die Dimensionen der Repräsentationen zu verringern. Nach den Conv-Blöcken werden die Einheiten aus den letzten Merkmalsabbildern (letzte Conv-Schicht) in einer eindimensionalen Schicht angeordnet, um die Informationen weiter durch die fc-Schichten propagieren zu lassen. Das originale VGG16-Netz verwendet zwei fc-Schichten mit jeweils 4096 Neuronen (Aktivierungsfunktion = $ReLU$), gefolgt von einer fc-Schicht für die Klassifizierung in 1000 Klassen (1000 Neuronen, Aktivierungsfunktion = $Softmax$). Weitere Informationen zur Architektur und den Parametern des Netzes sind in [3] und [30] zu finden. Trainiert wurde das VGG16-Netz auf der ImageNet-Datenbank [5], die aus mehreren Millionen RGB-Bildern[9] von Objekten aus 1000 verschiedenen Klassen besteht. Diese Bilder bieten eine große Variabilität, sodass das Netzwerk viele verschiedene Strukturen erlernen konnte, die unter Umständen auch nützlich sein könnten, um die gegebenen CT-Bilder zu klassifizieren.

Durch die feste Anzahl der Neuronen in den fc-Schichten und die dadurch ebenso festgelegte Struktur der Gewichte, ist der Eingang des originalen VGG16-Netzes auf RGB-Bilder mit 224×224 Pixeln festgelegt, wenn die trainierten fc-Schichten mitbenutzt werden sollen. Da für die Klassifikation der CT-Bilder in der letzten Schicht aber nur ein Neuron gebraucht wurde, das Werte zwischen null und eins annehmen kann, wurde das VGG16-Netz ohne die oberen drei fc-Schichten genutzt. Stattdessen wurden zwei selbst definierte fc-Schichten an die Conv-Blöcke angefügt, auf die sich das Training beschränken sollte. Die erste Schicht beinhaltete 256 Neuronen mit der $ReLU$-Funktion als Aktivierung und die zweite bestand aus einem Neuron, für das als Aktivierungsfunktion eine Sigmoidfunktion[10] verwendet wurde. Die von den vortrainierten Conv-Blöcken durch Lerntransfer erzeugten Repräsentationen sollten genutzt werden. Durch die erste selbsttrainierte fc-Schicht wurden daraus 256 Bildmerkmale erzeugt, welche die Grundlage für die Klassifikation in der letzten Schicht darstellten. In Folge dessen konnte die Dimension des Eingangs (Höhe und Breite) selbst gewählt werden. Nur die Tiefe der Bilder (Kanalanzahl = 3), musste bestehen bleiben, damit die Conv-Blöcke verwendet werden konnten. Zur Veranschaulichung ist der Aufbau des Netzes in Abbildung A.5 schematisch dargestellt. Die Dimension des Eingangs wurde auf $100 \times 100 \times 3$ festgelegt (drei Kanäle: RGB). Der Grund für die Wahl der Höhe und Breite wird in Kapitel 3.4 dargelegt. Für die speziellen Netzparameter wie Schicht- und Neuronenzahl in den fc-Schichten wurde sich aufgrund von Trainingstests entschieden. Wegen des geringen Datenumfangs in der Trainingskohorte sollten durch die zusätzlichen Schichten möglichst wenig anzupassende Gewichte erzeugt werden.

[9] Bilder mit drei Farbkanälen: R=Rot, G=Grün, B=Blau
[10] $f(x) := \frac{1}{1+\exp(-x)}$

3.4 Vorprozessierung der CT-Bilder

Damit die CT-Bilder der Patienten vom Netzwerk verarbeitet werden konnten, mussten die Bilder in einem mehrstufigen Prozess entsprechend vorbereitet werden. Zunächst wurden die DICOM-Dateien ins NIfTI-Format umgewandelt. Dabei wurden die Bilder auf ein neues Gitter mit einem einheitlichen Voxel-Volumen von $(1 \times 1 \times 1)\,\mathrm{mm}^3$ interpoliert. Dieser Schritt war notwendig, da in verschiedenen Krebsforschungszentren unterschiedliche Parameter der CT-Aufnahme genutzt wurden. So variierten zum Beispiel die Schichtdicken der CTs zwischen 3 und 5 mm, was durch die neue Aufteilung vereinheitlicht wurde. Die resultierenden 3D-CT-Bilder wurden durch quaderförmige Blöcke repräsentiert, bestehend aus Zellen (Voxel), die jeweils durch eine reelle Zahl in HOUNSFIELD-Einheiten repräsentiert wurden. Die 3D-Tumormasken bestanden aus genau so vielen Voxel, wie das dazugehörige CT-Bild. Jedoch nahmen die Werte der Voxel nur null (kein Tumor) oder eins (Tumor) an. Aus diesen Blöcken konnten Schichten extrahiert werden, die in Form einer (Bild-) Matrix dargestellt wurden. Dabei wurden die Schichten in Axial[11]-, Koronal[12]- und Sagittalschichten[13] unterteilt.
Für die Wahl der Prozessierungsschritte wurde sich an Radiomics-Studien wie zum Beispiel [17, 20, 21, 29] orientiert. Zur Überprüfung der Tumormaskierung wurde die Tatsache verwendet, dass Tumoren im Kopf-Hals-Bereich aus Weichteilgewebe mit CT-Zahlen zwischen -150 und 180 HE bestehen. Im CT-Bild wurde ermittelt, ob die vom Radioonkologen angefertigte Tumormaske Voxel einschließt, die Werte außerhalb dieses Bereichs aufweisen. Wenn dies der Fall war, wurden in der Tumormaske die entsprechenden Voxel von Eins auf Null gesetzt (Abb. 3.1). Damit sollten die menschlichen Fehler in der Tumormaskierung, die durch subjektive Eindrücke entstehen, zum Teil korrigiert werden. Anschließend wurde in jeweils axialer, koronaler und sagittaler Richtung die Schicht aus dem CT-Bild extrahiert, welche die größte Tumorfläche beinhaltete. So wurden die Schichtbilder mit den meisten Tumorinformationen gewonnen. Zusätzlich wurde die gleiche Schicht aus der Tumormaske extrahiert.
Das Betrachtungsfenster der CT-Zahlen wurde auf -200 bis 200 HE gesetzt, da die für die Charakterisierung interessanten Tumorstrukturen in diesem Bereich liegen. Mit eingerechnet ist ein gewisser Randbereich, sodass auch Randregionen und Gewebe mit ähnlicher Dichte unterscheidbar dargestellt werden können. Alle Voxelwerte größer als 200 HE wurden auf einen konstanten Wert von 200 HE und alle Werte kleiner als -200 HE auf -200 HE gesetzt. Dadurch wird die Grauwertskala gespreizt und der Kontrast für unterschiedliche Weichteilgewebe erhöht. Gleichzeitig wurden die Grauwerte auf eine Skala zwischen 0 und 255 reskaliert, da das VGG16-Netz auf 8-Bit RGB-Bildern trainiert wurde und somit die Gewichte auf Pixelwerte zwischen 0 und 255 angepasst sind (Abb. 3.1). Dies wurde für jedes Bild gleichermaßen realisiert, indem die Zuordnung $-200\,\mathrm{HE} \rightarrow 0$ und $200\,\mathrm{HE} \rightarrow 255$ angewendet und der gesamte

[11] Transversalebene bzw. Horizontalebene (Ebene senkrecht zur Längsachse des stehenden Menschen), erstreckt sich von links nach rechts und ventral nach dorsal
[12] Frontalebene, erstreckt sich von links nach rechts und kranial nach kaudal
[13] Ebene erstreckt sich von ventral nach dorsal und kranial nach kaudal

Bereich zwischen −200 und 200 HE linear zwischen 0 und 255 reskaliert wurde (Glg. (A.12)). Da die CT-Zahlen für jedes Bild durch die gleiche Zuordnung reskaliert wurden, blieben die physikalischen Informationen erhalten.

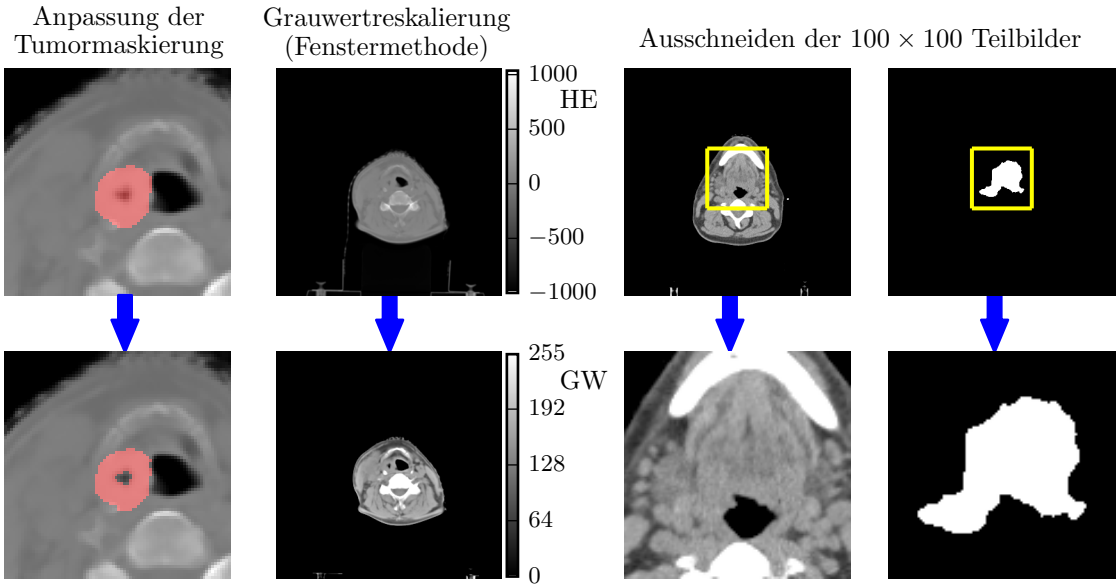


Abbildung 3.1: Vorprozessierungsschritte an axialen CT-Schnittbildern. Oben ist jeweils das Bild vor und unten nach dem Prozessierungsschritt dargestellt. Die Bilder zur Anpassung der Tumormaskierung (rot dargestellt) wurden vergrößert (ganz links). GW: Grauwerte, HE: HOUNSFIELD-Einheiten.

Schließlich wurden aus den resultierenden CT- und Tumormasken-Schichtbildern Teilbilder mit einer Dimension von 100 × 100 Pixel ausgeschnitten. Dafür wurde ein an den Tumor oben, unten, links und rechts angrenzender Kasten in das Bild gelegt. Die Seiten des Kastens wurden symmetrisch nach außen verschoben, bis sie einen Abstand von 100 Pixel zur gegenüberliegenden Seite erreichten. Der so gewonnene quadratische Bereich wurde ausgeschnitten und bildete das Teilbild (Abb. 3.1). Für den Fall, dass der Tumor in dem Schichtbild nicht komplett auf das 100 × 100 Teilbild passte, wurde der Flächenmittelpunkt des Tumorbereichs ermittelt und das Teilbild zentriert um diesen Punkt ausgeschnitten. Die Abmessung wurde gewählt, nachdem auf den Schichtbildern eine Statistik bezüglich der Seitenlängen der angrenzenden Kästen erstellt wurde. Daraus ging hervor, dass mehr als 95 % der Tumoren durch ein Teilbild in dieser Größe komplett erfasst wurden. Die Größe des Teilbildes sollte möglichst klein gewählt sein, um die Anzahl der trainierbaren Paramter im Deep-Learning-Netz gering zu halten. Dennoch sollten auch viele Tumoren vollständig durch das Teilbild erfasst werden, um die maximal möglichen Informationen in das Netzwerk geben zu können. Dies wurde durch einen Bereich der Größe 100 × 100 Pixel gewährleistet, der ebenfalls die Eingangsdimensionen des trainierbaren Netzes bestimmte.

Im letzten Schritt der Vorprozessierung wurden aus den ausgeschnittenen Teilbildern verschiedene Datensätze erstellt. Der Hintergrund dafür ist, dass das modifizierte neuronale Netz nur

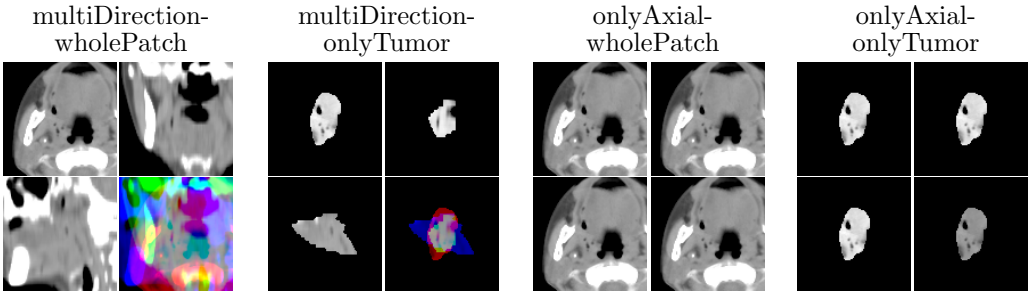


Abbildung 3.2: Aufbau bzw. Bildzusammensetzung der verschiedenen Datensätze. Links oben ist jeweils der R-Kanal, rechts oben der G-Kanal, links unten der B-Kanal und rechts unten das fertige RGB-Bild zu sehen.

RGB-Bilder mit drei Farbkanälen verarbeiten kann, die aber aus den vorhanden Teilbildern erst „zusammengebaut“ werden müssen. Zu jeder Kohorte (LRC-Training/ -Validierung, OS-Training/ -Validierung) wurden jeweils vier verschiedene Datensätze konstruiert, aus denen die beiden mit den besten Trainingsergebnissen für die Bildmerkmal-Extraktion ausgewählt wurden. Für den ersten Datensatz („MultiDirection-wholePatch“, Abb. 3.2) wurden die Bilder so erstellt, dass das axiale Teilbild als R-Kanal, das koronale Teilbild als G-Kanal und das sagittale Teilbild als B-Kanal genutzt wurde („MultiDirection“). Außerdem wurde jeweils das komplette Teilbild verwendet, sodass auch alle Regionen um den Tumor dargestellt wurden („wholePatch“). Die verschiedenen Richtungsteilbilder eines Tumors in einem Bild zu mischen, sollte dazu dienen, die maximal möglichen Informationen über den Tumor übergeben zu können. Das ganze Teilbild wurde genutzt, da somit auch Informationen über die Umgebung, also die Position des Tumors, erfasst wurden. Im zweiten Datensatz („MultiDirection-onlyTumor“, Abb. 3.2) wurden die Teilbilder mit ihrem Tumormaskenbereich gefaltet, bevor sie in die Bildkanäle gesetzt wurden. Das heißt, die Matrixeinträge wurden elementweise multipliziert. Da der Tumormasken-Bereich nur Einsen in den Pixeln hat, die im CT-Bild den Tumor darstellen und die restlichen Pixel Null sind, wird die gesamte Region im CT-Bild, die sich um den Tumor herum befindet, auf Null gesetzt („onlyTumor“). Dadurch werden mehr Informationen über die eigentliche Tumorgeometrie übergeben und das Netz bekommt keine unter Umständen irreführenden Informationen aus anderen Geweben. Der dritte Datensatz („onlyAxial-wholePatch“, Abb. 3.2) verwendet wieder das gesamte Teilbild mit den Regionen um den Tumor herum. Dafür werden alle drei Farbkanäle mit demselben axialen Teilbild belegt („onlyAxial“), da die axiale Richtung im CT-Bild die höchste Auflösung besitzt. Dadurch entsteht ein normales Schwarz-Weiß-Bild, in dem die Farbkanäle nicht gegeneinander verschoben sind. Der letzte Datensatz („onlyAxial-onlyTumor“) verbindet die Methode, in der nur axiale Teilbilder die Farbkanäle belegen mit der, in welcher die Regionen um den Tumor herum auf Null gesetzt werden (Abb. 3.2). Zum Schluss wurde die Reihenfolge der Patientendatensätze innerhalb einer Kohorte noch randomisiert, um ein allgemeineres Training zu ermöglichen.

3.5 Netzwerktraining

Die erstellten Datensätze wurden benutzt, um das modifizierte Netzwerk zu trainieren und die Trainingsergebnisse zu evaluieren. Die Gewichte der fünf Conv-Blöcke aus dem VGG16-Netz wurden dabei fest mit den Werten aus dem Vortraining besetzt und nicht weiter optimiert. Somit wurden nur die zufällig initialisierten Gewichte trainiert, durch welche die Neuronenwerte der letzten beiden fc-Schichten berechnet wurden. Bevor die Daten in das Netzwerk gegeben wurden, wurden sie durch ein mittleres Pixel normalisiert. Das heißt, es wurden die Werte aller Pixel über die gesamte Trainingskohorte gemittelt, sodass für jeden Farbkanal ein Mittelwert erzeugt wurde. Dieser Mittelwert wurde im entsprechenden Kanal von allen Pixeln der Bilder, die das Netzwerk passieren sollten, subtrahiert. Auch die Bilder der Validierungskohorte wurden mit denselben Werten normalisiert. Dieser Schritt war nötig, da Teile des vortrainierten VGG16-Netzes verwendet wurden und die Entwickler für ihr Training dieselbe Methode verwendeten [30].
Das Netz wurde über 250 Epochen trainiert, wobei die stochastische Optimierungsmethode Adam [15] gewählt wurde. Diese Methode zeigte oft bessere Leistungen und schnellere Konvergenz als andere Optimierungsmethoden wie SGD, was in den ersten eigenen Testversuchen bestätigt werden konnte. Die Teildatensätze, die pro Trainingsschritt in das Netzwerk gegeben wurden, bestanden aus 25 Bildern samt Risikogruppenzuordnung. Als Loss-Funktion wurde die Binäre-Kreuzentropie verwendet. Die Kreuzentropie H einer nach P verteilten, diskreten Zufallsvariable X und der Verteilung Q berechnet sich nach [4],

$$H(X, P, Q) = -\sum_{x} P(X = x) \cdot \log_2(Q(X = x)) \ . \tag{3.1}$$

Hier repräsentiert $x \in \{0, 1\}$ die Risikogruppenzuordnung eines Patienten und P gibt die Wahrscheinlichkeit dafür an, dass der Patient in der jeweiligen Risikogruppe ist. Da jeder Patient nur einer Risikogruppe zugeordnet werden kann und diese als wahr angenommen wird, gilt für jeden Patienten $P \in \{0, 1\}$. Der Netzwerkausgang Q stellt die vorhergesagte Wahrscheinlichkeit dafür dar, dass der Patient in Gruppe 1 ist. Dieser Wert liegt im Intervall $[0, 1]$, da die Aktivierungsfunktion des letzten Neurons durch die logistische Sigmoidfunktion beschrieben wird. Die vom Netz vorhergesagte Wahrscheinlichkeit, dass der Patient in Gruppe 0 ist, ergibt sich durch $1 - Q$.
Die Lernrate während des Trainings betrug 0,001 und entsprach somit dem Wert, der für die Optimierungsmethode Adam empfohlen wird [15]. Um einer Überanpassung des Modells entgegen zu wirken, wurde für die Gewichte zwischen den beiden fc-Schichten eine Dropoutrate von 0,7 verwendet. Das heißt, jedes betroffene Gewicht wurde in den einzelnen Trainingsschritten mit einer Wahrscheinlichkeit von 70 % bei der Anpassung vernachlässigt, um zu verhindern, dass sich das Netzwerk zu schnell an die Trainingsdaten anpasst und dadurch seine Verallgemeinerungsfähigkeit verliert. Zusätzlich wurde ein „Lernratenzerfall“ mit dem Parameter 0,01

verwendet, was bedeutet, dass die Lernrate ε nach jeder Epoche verringert wurde. Dieser Prozess wird durch Gleichung (3.2) beschrieben, in der ε_0 die Anfangslernrate, K die Anzahl der trainierten Epochen und λ der Zerfallsparameter ist,

$$\varepsilon = \varepsilon_0 \left(\frac{1}{1 + \lambda \cdot K} \right) \ . \tag{3.2}$$

Durch ε wird verhindert, dass sich die Parameter durch zu große Lernschritte wieder vom Minimum der Loss-Funktion entfernen, wenn sie schon gut optimiert waren.
Nach jeder Trainingsepoche wurde die Klassifizierungsleistung des Netzwerks evaluiert. Dafür wurde das Netz auf die gesamte Trainings- und Validierungskohorte angewendet und die Genauigkeit Acc[14] sowie die Fläche unter der Receiver-Operating-Characteristic(ROC)-Kurve AUC[15] der Klassifikation berechnet. Die Genauigkeit gibt den relativen Anteil der richtig klassifizierten Patienten (Schwellenwert der Klassifikation: 0,5) an. Die ROC-Kurve trägt die Sensitivität (Richtig-Positiv-Rate, TP) über der Falsch-Positiv-Rate (FP) der Klassifizierung auf. Dazu werden alle verschiedenen Netzwerkausgangswerte y ($y \in [0, 1]$) als Schwellenwerte für eine Klassifikation genutzt und bezüglich dieser Schwellenwerte die Richtig- und Falsch-Positiv-Rate gemäß

$$TP = \frac{r_\mathrm{p}}{r_\mathrm{p} + f_\mathrm{n}} \quad , \qquad FP = \frac{f_\mathrm{p}}{f_\mathrm{p} + r_\mathrm{n}} \tag{3.3}$$

ermittelt. Wenn Gruppe 1 als positiv angesehen wird, dann ist r_p die Anzahl der Patienten, die richtig zur Gruppe 1 eingeteilt wurden und f_n die Anzahl derer, die fälschlich zur Gruppe 0 gezählt wurden. Die Anzahl der Patienten, die fälschlich zur Gruppe 1 gezählt wurden, wird mit f_p bezeichnet und mit r_n die, welche richtig in Gruppe 0 eingestuft wurden. Daraus ergibt sich eine Kurve, die von der Diagonalen im TP-FP-Diagramm für eine zufällige Klassifikation immer weiter in die linke obere Ecke „gezogen“ wird, wenn die Klassifikation besser wird. Dabei ist der Anfangs- und Endpunkt (0,0) und (1,1) fest. Das heißt, AUC ist ein Maß für die Güte der Klassifikation, wobei $AUC = 0{,}5$ eine rein zufällige Klassifikation und $AUC = 1$ eine perfekte Klassifikation repräsentiert [39]. Am Ende des kompletten Trainings wurden die Gewichte des Netzwerks abgespeichert, damit das trainierte Netzwerk für Klassifikations- oder Bildmerkmal-Extraktions-Aufgaben verwendet werden konnte.

3.6 Bildmerkmal-Extraktion und Radiomics-Modellierung

Auf Grundlage der Trainingsverläufe mit den vier verschiedenen Datensätzen wurden die beiden „onlyAxial“-Datensätze für die weiteren Betrachtungen ausgewählt (Details Abschnitt 4.1). In anderen Deep-Learning-basierten Radiomics-Studien wurden die neuronalen Netze häufig

[14] für engl. Accuracy
[15] Area Under the Curve

nicht für die direkte Klassifikation der Daten verwendet, sondern um Deep-Bildmerkmale aus den Bildern zu extrahieren [17, 29]. Diese können durch konventionelle Radiomics-Methoden (Kap. 2.3), wie zum Beispiel Bildmerkmal-Selektion, weiterverarbeitet werden, um eine Signatur zu erstellen. Anhand dieser kann schließlich die Klassifikation vorgenommen werden. Da Deep-Bildmerkmale bereits prognostische Eigenschaften gezeigt haben, wurden auch aus den ausgewählten Daten Bildmerkmale mithilfe des CNNs extrahiert und mit konventionellen Machine-Learning-Methoden Radiomics-Modelle konstruiert [17, 22, 29].
Für alle Kohorten (LRC-Training/ -Validierung, OS-Training/ -Validierung) wurden aus dem „onlyAxial-wholePatch"- und dem „onlyAxial-onlyTumor"-Datensatz Bildmerkmale extrahiert. Der Vergleich der Vorhersagekraft zwischen „wholePatch"- und „onlyTumor"-Bildmerkmalen soll darüber Aufschluss geben, welche Informationen von größerer Bedeutung sind, die aus der Tumorumgebung oder die über die Tumorgeometrie. Für jeden Patienten wurde die vom trainierten Netz erstellte Repräsentation in der vorletzten fc-Schicht als 256-dimensionaler Bildmerkmalvektor abgespeichert (256 ST[16]-Deep-Bildmerkmale). Außerdem wurden in Anlehnung an [17] und [29] 4096 TL[17]-Deep-Bildmerkmale aus der vorletzten fc-Schicht des untrainierten, originalen VGG16-Netzes extrahiert. Diese 4096 Bildmerkmale wurden durch reinen Lerntransfer erzeugt. Um die Bilder aus den Datensätzen in das originale VGG16-Netz geben zu können, wurde vor der Pixelnormalisierung (Kap. 3.4) die Größe von $100 \times 100 \times 3$ auf $224 \times 224 \times 3$ geändert. Dafür wurde eine bikubische Interpolation genutzt. Um die prognostische Güte der verschiedenen Bildmerkmal-Sorten am Ende vergleichen zu können, wurden zusätzlich konventionelle Radiomics-Bildmerkmale ausgewertet. Diese wurden aus den vollständigen 3D-CT-Bildern der Patienten extrahiert. Mehr Details zu dieser Sorte von Bildmerkmalen sind in [20] zu finden.
Mit den extrahierten Bildmerkmalen wurden schließlich Radiomics-Modelle für die Klassifizierung der Patienten gebaut. Dazu wurde ein in der Arbeitsgruppe für Modellierung und Biostatistik in der Radioonkologie (OncoRay) entwickeltes Radiomics-Grundgerüst verwendet [20]. Dieses produziert Radiomics-Signaturen und optimiert Hyperparameter[18] von Machine-Learning-Algorithmen, um vorhersagekräftige Modelle zu trainieren [20]. Damit das beste Modell gefunden werden kann, bildet es Kombinationen aus einer Bildmerkmal-Selektions- und einer Klassifikationsmethode. Es wurden vier Machine-Learning-Methoden (Klassifikationsmethoden) benutzt: Logistische Regression (LogReg), naive Bayes (naiveBayes), Random Forest (RF) und eine Support Vector Machine (SVM). Die fünf verwendeten
Bildmerkmal-Selektionsmethoden waren Spearman-Korrelation (Spearman), Mutual Information Feature Selection (MIFS), Mutual Information Maximisation (MIM), Minimum Redundancy Maximum Relevance (MRMR) und Random Forest Variable Importance (RFVI). Eine kurze Beschreibung der einzelnen Methoden ist im Anhang A.8 zu finden. Jede der dar-

[16] selbst trainiert
[17] Transfer-Learning
[18] Parameter, die vor dem maschinellen Lernprozess gesetzt werden, diesen also beeinflussen

aus konstruierten zwanzig Kombinationen bildete ein Modell.
Um eine aussagekräftige Radiomics-Signatur zu erstellen, wurden als erstes alle Bildmerkmale der Trainingskohorte, die eine Varianz von null aufwiesen, verworfen. Die übrigen Bildmerkmale wurden in Gruppen zusammengefasst (SPEARMAN-Korrelationskoeffizient $> 0{,}90$) und aus jeder Gruppe nur ein repräsentatives Bildmerkmal weiter verwendet, um redundante Bildmerkmale auszuschließen. Anschließend wurde die jeweilige Bildmerkmal-Selektionsmethode zur Identifizierung der relevantesten Bildmerkmale auf 1000 Bootstraps[19] der Trainingskohorte angewendet. Die aus einem Bootstrap ausgewählten Bildmerkmale wurden in einer Rangfolge bezüglich eines von der Selektionsmethode abhängigen Zahlenwerts angeordnet. Danach wurden die Bildmerkmale j aus allen Bootstraps zusammengeführt und eine Rangliste nach ihrer Wichtigkeit I_j erstellt. Die Wichtigkeit wurde durch Gleichung (3.4) berechnet, worin R_{ij} der Rang des Bildmerkmals j im i-ten Bootstrap und occ_j die Auftrittfrequenz des Bildmerkmals über alle Bootstraps ist [20],

$$I_j = \frac{\sum_{i=1}^{1000} \sqrt{R_{ij}}}{occ_j^2} \ . \tag{3.4}$$

Dabei haben die wichtigsten Bildmerkmale die kleinsten I_j. Anschließend wurden die Hyperparameter der Machine-Learning- bzw. Klassifikationsmethode optimiert. Hyperparameter sind zum Beispiel Signaturgröße oder methodenspezifische Parameter [20]. Dazu wurde das Minimum einer Loss-Funktion in einem vordefinierten Parameterraum mittels einer zweifachen internen Kreuzvalidierung auf der Trainingskohorte mit 40 Wiederholungen ermittelt. Das heißt, pro Wiederholung wurde die Trainingskohorte in zwei Datensätze geteilt, mit denen abwechselnd trainiert und validiert wurde. Nähere Details (z.B. zur Loss-Funktion) sind in [20] zu finden.
Das aus Signatur und optimierter Machine-Learning-Methode bestehende Modell wurde mit 1000 Bootstraps der Trainingskohorte trainiert. Am Ende wurde die Klassifizierungsleistung des Modells auf der unabhängigen Validierungskohorte überprüft. Um die Modelle zu bewerten, wurde jeweils der *AUC*-Wert berechnet. Dabei wurde als kontinuierlicher Parameter die vom Modell vorhergesagte Wahrscheinlichkeit für die Risikogruppenzugehörigkeit eines Patienten genutzt. Der schematische Arbeitsablauf ist in Abbildung A.6 zusammengefasst.

Eine andere Methode zur Modellbewertung ist die KAPLAN-MEIER-Analyse. Bei der KAPLAN-MEIER-Analyse werden die Patienten auf Grundlage der vom Modell vorhergesagten Risikowahrscheinlichkeit stratifiziert. Das Modell liefert für jeden Patienten eine Wahrscheinlichkeit, dass er sich in der Hochrisiko- oder der Niedrigrisikogruppe befindet. Es wird ein Schwellenwert festgelegt, nach dem die Patienten in die entsprechenden Gruppen eingeteilt werden. Für jede Gruppe wird die Wahrscheinlichkeit, dass das betrachtete Ereignis nicht eintritt, über der

[19] eine Anzahl zufälliger Ziehungen von Stichproben aus einem Datensatz mit Zurücklegen (hier: genauso viele, wie Anzahl der Daten im Satz)

kontinuierlichen Ereigniszeit aufgetragen. Je mehr Zeit vergeht, desto wahrscheinlicher wird, dass das Ereignis eintritt. Dadurch sinkt die KAPLAN-MEIER-Kurve mit der Zeit ab. Konnten die Gruppen gut stratifiziert werden, sinkt die Kurve der Niedrigrisikogruppe langsamer als die der Hochrisikogruppe, sodass ein signifikanter Unterschied besteht.

4 Ergebnisse

In Abschnitt 4.1 sind die Ergebnisse des Netzwerktrainings dargestellt, das darauf abzielte, die Klassifikation mithilfe des CNNs von Anfang bis Ende durchzuführen. Die Ergebnisse der Modelle, die auf mathematisch definierten Bildmerkmalen (konventionelles Radiomics) oder auf Bildmerkmalen des CNNs basierten (Deep-Radiomics), sind in Abschnitt 4.2 dargestellt.

4.1 Training des modifizierten VGG16-Netzes

Das Netzwerk wurde für beide Therapieendpunkte mit der entsprechenden Trainingskohorte trainiert (LRC-/OS-Training) und mit der entsprechenden Validierungskohorte (LRC-/OS-Validierung) validiert. Zu jedem Trainingsverlauf wurde die Genauigkeit *Acc*, der Wert der Loss-Funktion *Loss* sowie die *AUC* nach jeder Trainingsepoche *N* auf beiden Kohorten (Trai-

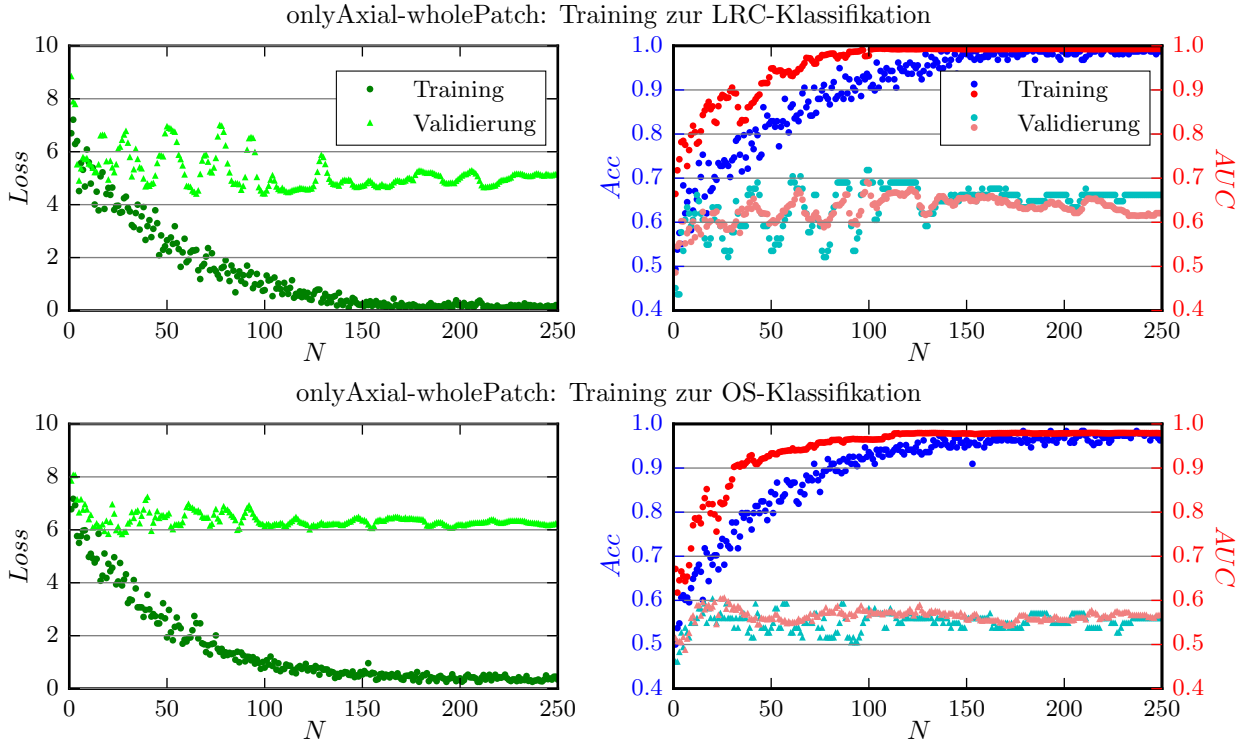


Abbildung 4.1: „onlyAxial-wholePatch"-Datensatz (Training und Validierung) für lokoregionäre Tumorkontrolle (LRC, oben) und Gesamtüberleben (unten). Gezeigt ist die Loss-Funktion (links) sowie die Genauigkeit *Acc* und die *AUC* (rechts) nach jeder Trainingsepoche *N*.

ning/Validierung) ausgewertet. Für das Training selbst, diente jedoch lediglich die Trainingskohorte. Als Beispiele sind die Verläufe der Trainings für die LRC- sowie die OS-Vorhersage mit den „onlyAxial-wholePatch"-Datensätzen in Abbildung 4.1 dargestellt. Die anderen Trainingsverläufe mit anderen Datensatzkonfigurationen sind im Anhang A.9 dargestellt. Die Ergebnisse, welche nach der letzten Trainingsepoche bestimmt wurden, sind in Tabelle 4.1 zusammengetragen.

Tabelle 4.1: Ergebnisse der Netzwerktrainings. Trainiert wurden die letzten beiden Schichten des modifizierten VGG16-Netzes für die Vorhersage der Risikowahrscheinlichkeit von zwei verschiedenen Therapieendpunkten. Dazu wurden unterschiedliche Datensatzkonfigurationen genutzt. Die Klassifizierungsleistung wurde am Ende des Trainings (nach 250 Epochen) unabhängig voneinander auf der Trainings- (T) und Validierungskohorte (V) ausgewertet.

Endpunkt	**Datensatzkonfiguration**	*Acc*		*AUC*	
		T	V	T	V
LRC	MultiDirection-wholePatch	0,96	0,54	0,98	0,53
	MultiDirection-onlyTumor	0,78	0,49	0,91	0,47
	onlyAxial-wholePatch	0,99	**0,66**	0,99	**0,62**
	onlyAxial-onlyTumor	**1,00**	0,54	**1,00**	0,61
OS	MultiDirection-wholePatch	**0,99**	0,44	**1,00**	0,41
	MultiDirection-onlyTumor	0,96	0,54	**1,00**	0,53
	onlyAxial-wholePatch	0,98	**0,56**	0,98	**0,57**
	onlyAxial-onlyTumor	**0,99**	0,55	**1,00**	0,55

Die Vorhersage eines loko-regionären Rezidivs ist von größerer Bedeutung als die des Gesamtüberlebens. Da letzteres Ereignis auch viele andere Ursachen haben kann, ist die Vorhersage auf Grundlage der CT-Bilder nur schwer möglich. Deshalb werden im Folgenden überwiegend die Ergebnisse der LRC-bezogenen Klassifizierung betrachtet und die OS-Ergebnisse höchstens als Vergleich genutzt. Des Weiteren wird für die Bewertung einer Klassifizierungsleistung hauptsächlich der *AUC*-Wert berücksichtigt. Da der *Acc*-Wert nur den relativen Anteil der richtig klassifizierten Patienten angibt, kann es passieren, dass eine eigentlich schlechte Klassifikation trotzdem relativ gute Werte ($Acc > 50\,\%$) erreicht. So könnte das Netz, wenn es für die Klassifikation bezüglich LRC trainiert wurde, bei der Validierung eine Genauigkeit von 57,75 % einfach dadurch erreichen, dass es alle Patienten in Gruppe 1 einteilt. Dazu muss es nicht gelernt haben, die Patienten zu unterscheiden, sondern einfach alle in die eine Gruppe einzuteilen. Diese Tendenz ist bei der Bewertung durch den *AUC*-Wert nicht so stark ausgeprägt. Wenn das Netz für alle Patienten genau den Wert 1 produzieren würde, ergäbe sich eine *AUC* von 0,5, was einer rein zufälligen, also schlechten Klassifikation, entspricht.
Die Trainingsverlaufskurven für alle Datensätze zeigen qualitativ einen ähnlichen Verlauf (Abb. 4.1, A.7, A.8, A.9). Nur zu Trainingsbeginn unterscheiden sich die Kurven zum Teil.

So stieg die Genauigkeit und der *AUC*-Wert für den „MultiDirection-onlyTumor“-Datensatz auf der LRC-Trainingskohorte nicht so schnell wie bei den anderen Datensätzen. Wahrscheinlich wurde daher in den 250 trainierten Epochen auch nicht ein so hohes Ergebnis erreicht ($AUC = 0{,}91 < 0{,}98$). Außerdem ist in den Trainingsergebnissen (Tab. 4.1) zu sehen, dass die Klassifizierungsleistung für alle Datensätze auf den Trainingsdaten deutlich besser ist, als in der Validierung. Das trainierte Netzwerk erreicht auf den Trainingskohorten zum Teil *AUC*- und *Acc*-Werte von 1,0 (LRC-Training, onlyAxial-onlyTumor), während die beste Klassifikationsleistung bezüglich LRC in der Validierung $AUC = 0{,}62$ erreicht (onlyAxial-wholePatch). Einige Validierungsergebnisse haben sogar einen Wert von $AUC < 0{,}5$, was auf eine Prognose in die falsche Richtung hinweist. In der LRC-Klassifikation ist eine leichte Tendenz zu erkennen, dass die „onlyAxial“-Datensätze bessere Ergebnisse erreicht haben als die „MultiDirection“-Datensätze (Tabelle 4.1).

Diskussion der Trainingsergebnisse

Mit konventionellen Radiomics-Ansätzen wurden in der Vergangenheit für verschiedene Risikovorhersagen *AUC*-Werte von bis zu 0,7 erreicht [1, 20, 21]. Die Hypothese dieser Arbeit war, dass mit einem trainierten Deep-Learning-Netz bessere Ergebnisse in der Klassifikation erreicht werden könnten. Wider dieser Erwartung sind die erreichten Ergebnisse nur vergleichbar mit denen, die mittels durchschnittlich guten, konventionellen Radiomics-Modellen erreicht wurden [20].

Die vier mit unterschiedlichen Datensätzen und ansonsten gleichen Trainings- und Netzparametern durchgeführten Netzwerktrainings zeigen, dass das qualitative Klassifizierungsverhalten für alle Datensätze ähnlich ist. Die Unterschiede im Trainingsbeginn sind vermutlich auf die randomisiert initialisierten Gewichte der letzten beiden fc-Schichten zurückzuführen. Wenn diese gerade zufällig ungünstige Werte annehmen, kann das dazu führen, dass das Netzwerk nur schwer seine Parameter anpassen kann oder gar kein Lernprozess stattfindet. Das könnte die Ursache für den abweichenden Trainingsverlauf der „MultiDirection-onlyTumor“-Daten sein. Interessant ist jedoch, dass das Training mit den „onlyAxial“-Datensätzen tendenziell etwas besser funktioniert hat. Das zeigt, dass das auf normalen Bildern vortrainierte CNN wahrscheinlich die „MultiDirection“-Bilder nicht so gut „erkennen“ kann. In normalen Bildern wie Fotografien sind die Strukturen der einzelnen Farbkanäle voneinander abhängig. In den „MultiDirection“-Bildern stellen die Farbkanäle völlig unterschiedliche Strukturen dar. Das könnte ein Grund dafür sein, dass die vortrainierten Conv-Blöcke die Bilder nicht gut verarbeiten konnten.

Die niedrigen *AUC*- und *Acc*-Werte in der Validierung im Vergleich zu den hohen Werten im Training weisen auf eine starke Überanpassung der Modelle im Training hin. Das Netzwerk lernte in den 250 Epochen die Trainingsdaten „auswendig“, sodass nahezu alle Patienten aus der Trainingskohorte in die richtige Risikogruppe eingeordnet werden konnten (alle Datensätze der

Trainingskohorten, $AUC > 0{,}9$). Dabei erreichte es aber kaum Verallgemeinerungsfähigkeit. Neue Daten (Validierungsdaten) konnten nur schlecht klassifiziert werden, sodass die Validierungsergebnisse teilweise nur einer zufälligen Klassifikation entsprechen. Dies gilt sowohl für die Einordnung bezüglich des Rezidivrisikos (MultiDirection-wholePatch, $AUC = 0{,}53$), als auch bezüglich des Gesamtüberlebens (MultiDirection-onlyTumor, $AUC = 0{,}53$ und onlyAxial-onlyTumor, $AUC = 0{,}55$). Der Versuch, dies durch eine hohe Dropoutrate und möglichst wenige zu trainierende Parameter zu reduzieren, blieb leider erfolglos. Eventuell war die gewählte Dropoutrate von 0,7 sogar zu hoch, sodass das Training negativ beeinflusst wurde. Zum Beispiel könnte das starke Rauschen im Trainingsverhalten auch dadurch verursacht worden sein. Jedoch wurde Dropout nur zwischen den letzten beiden Schichten angewendet. Der Großteil der trainierbaren Parameter befand sich aber zwischen der letzten Conv-Schicht (die in eine eindimensionale Form gebracht wurde) und der ersten fc-Schicht, welche durch mehr als eine Million Gewichte verbunden werden. Es könnte also auch sein, dass der Dropout gar keinen großen Einfluss auf das Training hatte. Dieser Punkt sollte in neuen Untersuchungen unbedingt berücksichtigt werden, da die Parameterzahl offensichtlich zu groß für die Datenmenge von 158 Trainingspatienten (LRC-Training) war. Aus diesem Grund sind weitere Schritte für die Planung zukünftiger Experimente zu berücksichtigen. Da der qualitative Trainingsverlauf für alle Datensätze ähnlich aussieht, scheint der genaue Aufbau der Datensätze (Kanalbelegung, Teilbildextraktionsmethode, etc.) zumindest im bisherigen Stadium der Untersuchung eine untergeordnete Rolle zu spielen. Vermutlich können die Leistungen vorerst effektiver verbessert werden, indem die Netz- und Trainingsparameter und ihre Auswirkung auf das Training genauer untersucht werden. Leider blieb im Rahmen dieser Bachelorarbeit nicht genügend Zeit, um alle Parameter und Möglichkeiten ausreichend zu erforschen, weshalb die oben genannten Einstellungen verwendet wurden (Kap. 3.3 und 3.5). So lassen manche Trainingsverläufe (Abb. 4.1, A.8) zum Beispiel vermuten, dass über zu viele Epochen trainiert wurde. Außerdem ist dringend zu empfehlen, die Datensätze zu vergrößern, um eine bessere Trainingsgrundlage zu haben und so die Verallgemeinerungsfähigkeit des Netzes zu steigern. Solche zusätzlichen Daten können auch künstlich aus den vorhandenen Daten durch Rotationen, Translationen und Überlagerungen von Störsignalen (Rauschen) generiert werden. Auch andere Lernraten (z. B. niedrigere oder schneller abfallende) oder andere Optimierungsmethoden[1] könnten vielleicht das Training verbessern.

Eine weitere Ursache für die geringen AUC-Werte in der Validierung kann die Netzarchitektur selbst sein. Hier wurde das vortrainierte VGG16-Netz verwendet, das von Keras bereitgestellt wird. Dieses Netz ist viel tiefer als alle Netze, die in der Literatur für ähnliche Aufgaben vorgestellt wurden [17, 29]. Durch die vortrainierten Conv-Blöcke könnten schon zu abstrakte Deep-Bildmerkmale produziert worden sein, die keine aussagekräftige Repräsentation mehr darstellten. Um vorerst Ergebnisse wie in [17] oder [29] zu reproduzieren, sollten Experimen-

[1] z. B. auf `https://keras.io/optimizers/`

te mit ähnlich tiefen Netzen durchgeführt werden. Dafür bietet sich ein Wechsel des Deep-Learning-Grundgerüsts an, um Zugang zu anderen vortrainierten Netzen zu erhalten oder das eigenständige Trainieren eines passenden Netzes.
Das modifizierte VGG16-Netz konnte mit keinem der vier Datensätze so trainiert werden, dass Validierungsdaten exakt klassifiziert werden konnten. Mit dem Deep-Learning-Netzwerk allein war es also nicht möglich, die Risikogruppenzugehörigkeit von Kopf-Hals-Tumor-Patienten vorherzusagen. Dennoch könnten die generierten Deep-Bildmerkmale eine gewisse Vorhersagekraft besitzen. Auch in anderen Studien wurden nur die Deep-Bildmerkmale verwendet, was darauf hinweisen könnte, dass sich NNs allein noch nicht eignen, um Patienten anhand von medizinischen Bildern zu klassifizieren [17, 22, 29]. Da die Trainings- und Validierungskohorten der „onlyAxial"-Datensätze die höchsten AUC-Werte erzielten (Tabelle 4.1), wurden diese beiden Datensätze benutzt, um Deep-Bildmerkmale zu extrahieren (Kap. 3.6). Um ein valides Radiomics-Modell auf diese Weise zu konstruieren, sollten die Ergebnisse der Validierung nicht verwendet werden. Doch kann wie hier ein unabhängiger Datensatz benutzt werden, um einen ersten Eindruck der Verallgemeinerungsfähigkeit des Modells zu erhalten. Die eigentliche Validierung des Modells muss mit einem völlig unabhängigen, möglichst externem Datensatz durchgeführt werden.

4.2 Modellierung mit Radiomics-Grundgerüst

Die von den konventionell trainierten Modellen erreichten Leistungen bei der Klassifikation der Trainings- und Validierungskohorte wurden in Matrizen zusammengefasst. Exemplarisch ist dies in Abbildung 4.2 für die Modelle dargestellt, die für die Klassifikation bezüglich LRC mit ST-Deep-Bildmerkmalen[2] aus den „onlyAxial-wholePatch"-Datensätzen trainiert wurden. Darüber hinaus wurden die Modelle mit TL-Deep[3]- und konventionellen Radiomics-Bildmerkmalen (Kap. 3.6) trainiert. Die anderen Matrizen, welche die AUC-Werte aller Modelle beinhalten, sind im Anhang A.10 dargestellt.
Pro Klassifikationsaufgabe (LRC und OS) wurde für jede Bildmerkmal-Sorte (ST-Deep-wholePatch, TL-Deep-wholePatch, ST-Deep-onlyTumor, TL-Deep-onlyTumor, Radiomics) ein Modell ausgewählt. Es wurden lediglich Modelle gewählt, die als Lernmethode die logistische Regression verwendeten, da diese eine Standardmethode ist, die leicht zu verstehen ist und oft angewendet wird. Zur Auswahl der passenden Bildmerkmal-Selektions-Methode wurde für jede dieser Methoden der Median über die AUC-Werte (Training) aller Lernmethoden gebildet. Mit der Wahl der Methode mit dem höchsten medianen AUC-Wert sollte ein möglichst gutes, aber auch repräsentatives Modell gewählt werden. Die zu den verschiedenen Klassifikationsaufgaben und Bildmerkmal-Sorten gewählten Modelle sind in Tabelle 4.2 zusammen mit ihren

[2]Bildmerkmale aus dem selbst trainierten (ST) CNN
[3]Bildmerkmale aus dem originalen VGG16-Netz, durch Lerntransfer (Transfer-Learning, TL) erzeugt

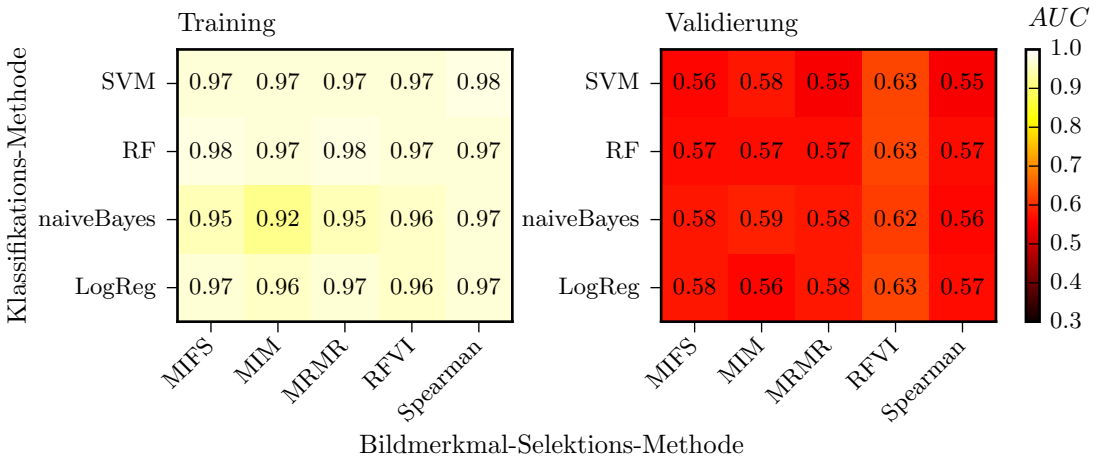


Abbildung 4.2: Beispielergebnisse der Modellierungen mit dem Radiomics-Grundgerüst. Die Modellierung wurde mit den 256 ST-Deep-Bildmerkmalen aus dem „onlyTumor-wholePatch"-Datensatz vorgenommen, um die Risikogruppe bezüglich eines loko-regionären Rezidivs vorherzusagen. Die Spalten gehören jeweils zu einer Bildmerkmal-Selektions-Methode und die Zeilen zu einer Klassifikations- bzw. Machine-Learning-Methode (Kap. 3.6).

Klassifizierungsleistungen dargestellt.

Auch beim konventionellen Training findet starke Überanpassung der Modelle unter Verwendung der Deep-Bildmerkmale statt. Auf der Trainingskohorte wurden sehr gute Klassifikationen erreicht (LRC: $AUC \geq 0{,}80$), während die Validierungskohorte nicht viel besser als zufällig in die Risikogruppen aufgeteilt wurde (LRC: $AUC \leq 0{,}62$). Die TL-Deep-Bildmerkmale, welche mit dem nicht trainierten VGG16-Netz mittels Lerntransfer produziert wurden, weisen eine geringere Überanpassung als die ST-Deep-Bildmerkmale auf, die aus dem modifizierten

Tabelle 4.2: Für unterschiedliche Klassifikationsaufgaben und verschiedene Bildmerkmale gewählte Modelle mit ihren Klassifizierungsleistungen auf der Trainings- (T) und Validierungskohorte (V).

Endpunkt	**Bildmerkmal-Sorte**	**Modell-kombination**	*AUC*	
			T	V
LRC	ST-Deep-wholePatch	MRMR-LogReg	0,97	0,58
	ST-Deep-onlyTumor	MRMR-LogReg	0,98	0,62
	TL-Deep-wholePatch	RFVI-LogReg	0,80	0,60
	TL-Deep-onlyTumor	MRMR-LogReg	0,86	0,50
	Radiomics (konventionell)	MIM-LogReg	0,76	0,65
OS	ST-Deep-wholePatch	RFVI-LogReg	0,97	0,53
	ST-Deep-onlyTumor	RFVI-LogReg	0,99	0,51
	TL-Deep-wholePatch	MIM-LogReg	0,82	0,55
	TL-Deep-onlyTumor	RFVI-LogReg	0,76	0,62
	Radiomics (konventionell)	Spearman-LogReg	0,72	0,59

Netz stammen. Dies spiegelt das Resultat der Überanpassung wider, die schon während des Deep-Learnings stattgefunden hat. Andererseits sind die Klassifizierungsleistungen in der Validierung bei den TL-Deep-Bildmerkmalen nicht besser als bei den ST-Deep-Bildmerkmalen (Tabelle 4.2, LRC). Das heißt, auch bei etwas geringerer Überanpassung ist die Verallgemeinerungsfähigkeit und die damit verbundene Vorhersagekraft der auf TL-Deep-Bildmerkmalen trainierten Modelle nicht besser. Die konventionellen Radiomics-Bildmerkmale zeigen dagegen noch weniger Überanpassungseffekte bei ungefähr gleicher (OS: $AUC = 0{,}59$) und sogar besserer (LRC: $AUC = 0{,}65$) Klassifizierungsleistung auf der Validierungskohorte. Hier konnte also kein Vorteil von durch Deep-Learning generierten Bildmerkmalen gegenüber den konventionellen Bildmerkmalen gezeigt werden.

Für drei verschiedene LRC-Vorhersage-Modelle wurde eine Kaplan-Meier-Analyse durchgeführt: Für das MIM-LogReg-Modell, das mit konventionellen Radiomics-Bildmerkmalen erstellt und trainiert wurde, für das MRMR-LogReg-Modell, das mit ST-Deep-onlyTumor-Bildmerkmalen konstruiert und trainiert wurde und für das MRMR-LogReg-Modell, das mit TL-Deep-onlyTumor-Bildmerkmalen erstellt und trainiert wurde. Um für die drei gewählten Modelle den Schwellenwert für die Gruppeneinteilung festzulegen, wurden die vom Modell vorhergesagten Wahrscheinlichkeiten der Niedrigrisikogruppenzughörigkeit sowie die wahren Risikogruppenzugehörigkeiten der Patienten verwendet. Damit wurde eine ROC-Kurve konstruiert (Abb. 4.3, A.19), in welcher der Punkt bestimmt wurde, der am weitesten von der Diagonalen entfernt lag. Der Wahrscheinlichkeitswert, durch den dieser Punkt entstanden war, wurde als Schätzer für den optimalen Schwellenwert für die Gruppeneinteilung verwendet. Die optimalen Schwellwerte $\tilde{p}$ für die untersuchten Modelle sind in Tabelle 4.3 zusammengefasst.

Anhand des Schwellenwertes wurden die Patienten der Trainings- sowie Validierungskohorte mit ihrem Vorhersagewert des Modells in die Gruppen eingeteilt. Die daraus resultierenden

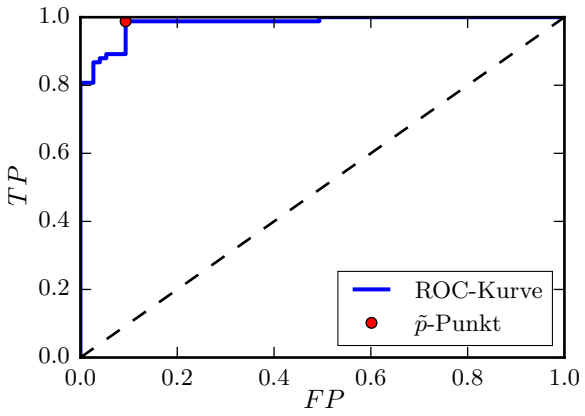


Abbildung 4.3: ROC-Kurve für die Klassifikation der Patienten mit dem MRMR-LogReg-Modell, das mit ST-Deep-onlyTumor-Bildmerkmalen trainiert wurde. Der $\tilde{p}$-Punkt ist am weitesten von der Diagonalen entfernt.

Tabelle 4.3: Optimale Wahrscheinlichkeitsschwellenwerte $\tilde{p}$ für die Einteilung der Patienten in die Hochrisikogruppe ($p \leq \tilde{p}$) oder die Niedrigrisikogruppe ($p > \tilde{p}$) bezüglich LRC.

Modell	**Bildmerkmal-Sorte**	$\tilde{p}$
MRMR-LogReg	ST-Deep, onlyTumor	0,25842
MRMR-LogReg	TL-Deep, onlyTumor	0,57871
MIM-LogReg	konvent. Radiomics	0,54306

Kaplan-Meier-Kurven sind in Abbildung 4.4 dargestellt. Es ist deutlich zu sehen, dass keine der beide Stratifizierungen der auf CNN-Bildmerkmalen basierten Modelle auf der Validierungskohorte signifikant war: $p > 0{,}05$. Das beste Modell war jenes, welches auf Grundlage der konventionellen Radiomics-Bildmerkmale erstellt wurde. Dieses erreichte als einziges eine

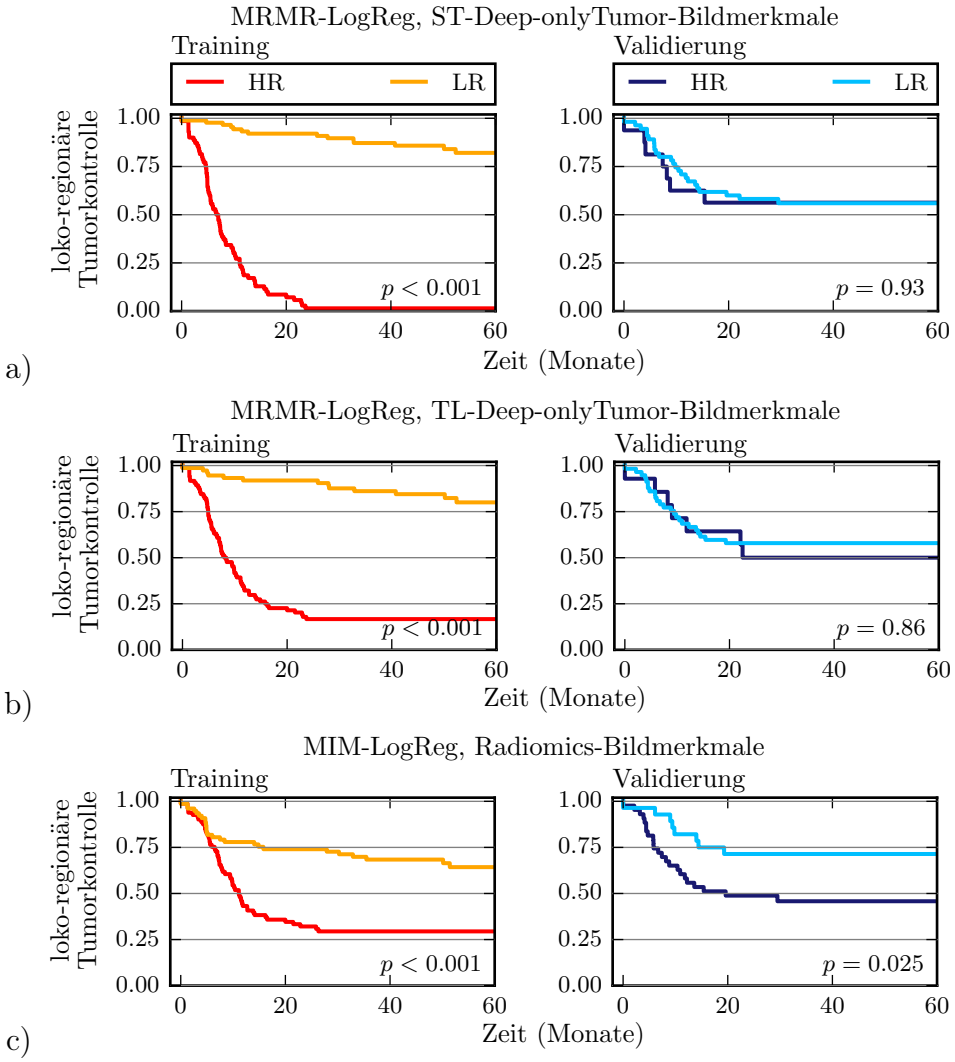


Abbildung 4.4: Kaplan-Meier-Stratifizierungen für loko-regionäre Tumorkontrolle mithilfe der bestimmten Schwellenwerte. Keine der Stratifizierungen durch CNN-Bildmerkmale ist auf der Validierungskohorte signifikant ($p > 0{,}05$, Log-Rank-Test). Einzig die konventionellen Bildmerkmale erreichen eine signifikante Stratifizierung ($p = 0{,}025$, Log-Rank-Test). HR: Hochrisiko, LR: Niedrigrisiko.

signifikante Stratifizierung $p = 0{,}025 < 0{,}05$. Auch dort sind Überanpassungseffekte zu erkennen, jedoch weniger ausgeprägt. Bei den Modellen der Deep-Bildmerkmale ist dagegen keine Unterscheidung der Gruppen in der Validierung möglich. Die Einteilung der Trainingskohorte hat dagegen sehr gut funktioniert ($p \leq 0{,}001$). Da die Modelle aber keine Allgemeingültigkeit besitzen, können sie nicht verwendet werden, um den Behandlungserfolg von Kopf-Hals-Tumorpatienten vorherzusagen.

Diskussion der Modellleistungen

Die Klassifikationsergebnisse der unterschiedlichen Modelle zeigen, dass mit den Deep-Bildmerkmalen, die aus den CNNs extrahiert wurden, keine guten prognostischen Radiomics-Modelle erstellt werden konnten. Zur Vorhersage des LRC-Risikos innerhalb von 24 Monaten nach Therapiebeginn war das Modell, das auf Grundlage der konventionellen Radiomics-Bildmerkmale konstruiert wurde, am besten geeignet ($AUC_{\text{valid}} = 0{,}65$). Auch bei der Klassifikation bezüglich des Gesamtüberlebens zeigte sich ein ähnliches Verhalten, da das Modell am geringsten überangepasst war. Hier erreichten drei der mit Deep-Bildmerkmalen erstellten Modelle sogar nur Werte von $AUC \leq 0{,}55$, was eine sehr zufällige Klassifikation repräsentiert. Etwas bessere Ergebnisse wurden in der Klassifikation bezüglich LRC erreicht. Die besten CNN-Bildmerkmale waren dort die ST-Deep-onlyTumor-Bildmerkmale, mit denen ein AUC-Wert von 0,62 in der Validierung erreicht wurde. Leider zeigte auch in der konventionellen Modellierung mit Deep-Bildmerkmalen die Überanpassung des Netzes negative Wirkung. Die Bildmerkmale aus dem selbst trainierten Netz erreichten fast perfekte Klassifikationsleistungen auf der Trainingskohorte ($AUC \geq 0{,}97$), während die Validierungskohorte nicht besser als mit $AUC = 0{,}62$ unterteilt werden konnte. Offensichtlich wurde durch das Netztraining die Repräsentation in der letzten fc-Schicht genau auf die Klassifikation der Trainingsdaten angepasst. Dadurch wurden aber mit neuen Daten eher zufällige Repräsentationen bezüglich der Klassifikation erstellt. Da die konventionellen Radiomics-Modelle ebenfalls mit genau den angepassten Bildmerkmalen trainiert wurden, fiel es dem Modell leicht, Unterscheidungskriterien in diesen Bildmerkmalen für die Klassifikation der Trainingsdaten zu finden. Daher konnten die Hoch- und Niedrigrisikogruppe gut getrennt werden. Die Patienten der Validierungskohorte konnten dagegen mit den gefundenen Unterscheidungskriterien nur schwer klassifiziert werden. Dies spiegelt sich in den Kaplan-Meier-Analysen der drei ausgewählten Modelle wider, in denen auf der Validierungskohorte nur mit den konventionellen Bildmerkmalen eine signifikante Stratifizierung der Risikogruppen erreicht werden konnte ($p = 0{,}025$).
Die TL-Deep-Bildmerkmale, welche durch reinen Lerntransfer erzeugt wurden, weisen ebenfalls Überanpassungseffekte auf. Zwar sind die AUC-Werte auf der Trainingskohorte nicht ganz so hoch ($< 0{,}90$) wie bei den ST-Deep-Bildmerkmalen ($> 0{,}95$), jedoch sind die Klassifizierungen auf der Validierungskohorte auch nicht höher. Bei der Risikogruppenvorhersage für LRC-Ereignisse erreichen die TL-Deep-onlyTumor-Bildmerkmale in der Validierung sogar das

niedrigste Ergebnis ($AUC = 0{,}50$). In der KAPLAN-MEIER-Stratifizierung für dieses Modell ist ebenfalls kein Unterschied zwischen den Kurven der Hoch- und Niedrigrisikogruppe erkennbar. Dies könnte unter anderem ein Anzeichen dafür sein, dass die verwendeten Deep-Bildmerkmale schon zu abstrakt sind und kaum noch Informationen enthalten, die für die Klassifizierung von Bedeutung sind.
Die in Tabelle 4.2 aufgelisteten Modelle sind nur repräsentative Modelle, die mittels logistischer Regression erzeugt wurden. Für andere Modelle wurden höhere AUC-Werte in der Validierung erzielt (Anhang A.10).
Wahrscheinlich resultiert die Überanpassung der Radiomics-Modelle direkt aus der Überanpassung des CNNs. Die Eingang-Ausgang-Funktion des neuronalen Netzes, die durch 158 Punkte (Trainingsdaten) definiert ist, wird mit mehr als einer Million Parametern modelliert, was zwangsläufig zu einer Überanpassung führt. Viele der 256 ST-Deep-Bildmerkmale aller Patienten haben den Wert null, sodass das neuronale Netz für die Klassifikation in der letzten fc-Schicht gar nicht auf alle Bildmerkmale aus der vorletzten Schicht angewiesen ist. Dies ist ein Anzeichen für das „Auswendig-Lernen“, wodurch die Eingang-Ausgang-Funktion ihre Allgemeingültigkeit verliert. Diese „null“-Bildmerkmale wirken sich natürlich auch auf die Modellierung mit den konventionellen Machine-Learning-Methoden aus, da aus ihnen keine Informationen gewonnen werden können. Bei einem Versuch, das Netzwerk zur Vermeidung der „null“-Bildmerkmale mit nur vier Neuronen in der vorletzten Schicht zu trainieren, konnten jedoch keine besseren Ergebnisse erzielt werden.

5 Zusammenfassung und Ausblick

In dieser Arbeit wurde untersucht, inwiefern sich Deep-Learning-Methoden eignen, um den Behandlungserfolg der Strahlentherapie von Kopf-Hals-Tumor-Patienten anhand ihrer CT-Bilder vorherzusagen (LRC- und OS-Risiko nach der Therapie). Die Patienten sollten maschinell in eine Hoch- oder Niedrigrisikogruppe eingeteilt werden. Dafür wurde ein neuronales Netz mittels Deep-Learning trainiert, wobei teilweise Lerntransfer verwendet wurde. Außerdem wurden durch konventionelle Machine-Learning-Methoden Radiomics-Modelle auf Grundlage von Deep-Bildmerkmalen für die Klassifizierung konstruiert. Aus den CT-Bildern, Tumormasken und Therapieausgängen von 302 Patienten wurden verschiedene Datensätze für das Training und die Validierung erstellt, aus denen mithilfe des verwendeten neuronalen Netzes auch die Deep-Bildmerkmale extrahiert wurden.

Die Vermutung, dass die Patientenklassifizierung mithilfe von Deep-Learning im Vergleich zu konventionellen Machine-Learning-Methoden verbessert werden könne, konnte im Rahmen dieser Arbeit nicht bestätigt werden. Das modifizierte und anschließend zum Teil trainierte VGG16-Netz erreichte mit den prozessierten CT-Bildern eine maximale Klassifikationsleistung von $AUC = 0{,}62$ in der Validierung (Klassifikation bezüglich LRC-Risiko). Dieser Wert stellt zwar keine zufällige Klassifikation mehr dar, jedoch werden mit konventionellen Radiomics-Methoden teilweise bessere Leistungen erzielt [20]. Mit der Extraktion von Deep-Bildmerkmalen, um damit auf konventionelle Weise Radiomics-Modelle zu erstellen, konnten auch keine besseren Ergebnisse in der Validierung erzielt werden. Das Modell zur LRC-Risiko-Vorhersage, das auf konventionell berechneten Radiomics-Bildmerkmalen basierte, erreichte sogar eine bessere Klassifizierungsleistung ($AUC = 0{,}65$) als die Deep-Bildmerkmal-basierten LRC-Modelle.

Das Hauptproblem der Deep-Learning-Ansätze war die Überanpassung des Netzes. Die Modelle konnten die Trainingsdaten nach dem Training fast perfekt einteilen, während die im Training nicht verwendeten Validierungsdaten zum Teil nur zufällig klassifiziert wurden (s. Kaplan-Meier-Stratifizierung). Der Hauptgrund dafür liegt vermutlich am geringen Umfang der Trainingsdaten. Vor allem für das neuronale Netz mit mehr als einer Million Parametern zur Modellierung der Eingang-Ausgang-Funktion reichte die verfügbare Anzahl an Patienten nicht aus, obwohl Versuche unternommen wurden, die Überanpassung durch Methoden wie Dropout oder Lernratenzerfall zu reduzieren. Dieses Phänomen setzte sich schließlich auch in den konventionellen Modellen fort, für die die Bildmerkmale aus dem Netz extrahiert wurden.

Das verwendete CNN (VGG16) ist sehr tief, sodass die errechneten Bildmerkmale der vorletzten Schicht möglicherweise schon zu abstrakt gewesen sind, um noch brauchbare Informationen zu enthalten. Das könnte auch ein Grund für die schlechten Ergebnisse der konventionell konstruierten Deep-Bildmerkmal-basierten Radiomics-Modelle sein.

Auch wenn in dieser Arbeit die Deep-Learning basierte Vorhersage von Behandlungserfolgen keine besseren Ergebnisse als die konventionelle Radiomics-Methode erzielte, hat Deep-Learning ein großes Potential. Dies wird in anderen Bereichen der maschinellen Bildverarbeitung deutlich [19]. Um dieses Potential im klinischen Kontext auszuschöpfen müssen die in dieser Arbeit aufgetretenen Probleme, z. B. Modellüberanpassung im Training, in zukünftigen Studien weiter reduziert werden. Es gibt viele Parameter, durch die das Netz und das Training definiert werden, welche großen Einfluss auf die erreichbare Leistung des neuronalen Netzes haben. Im Rahmen dieser Bachelorarbeit konnten diese Parameter nicht alle untersucht und optimiert werden, sie können aber Gegenstand weiterer Untersuchungen sein.
Eine essentielle Bedingung zur Erstellung valider Modelle ist die Vergrößerung des Trainingsdatensatzes. Der Patientendatensatz könnte zum einen durch Kooperationen verschiedener Kliniken erhöht werden, was zu einer höheren Heterogenität im Datensatz und somit einer besseren Verallgemeinerbarkeit der Resultate führen kann. Darüber hinaus können aus den vorhandenen Daten durch Bildmodifikationen künstlich neue Bilddaten erzeugt werden. Dadurch könnten Modelle allgemeiner trainiert und die Überanpassung reduziert werden. Des Weiteren muss unbedingt die Wahl der besten Deep-Learning- und Netzparameter für diese Art von Klassifizierungsaufgaben genauer untersucht werden. Es sollten verschiedene Netzwerkarchitekturen und ihre Einsatzfähigkeit für klinische Bildgebungsdaten betrachtet werden. Parameter wie die Anzahl der Schichten im Netz, deren Aufbau oder die Optimierungsmethode spielen dabei vermutlich eine große Rolle. Eine tiefere Untersuchung der Repräsentationen in den einzelnen Netzwerkschichten, deren Aufbau und die Verbindung zu Strukturen im Originalbild könnte ebenfalls hilfreich sein, um auch neuronale Netze speziell für die Anwendung auf medizinische Bilder zu konstruieren.
Ein besseres Verständnis der Deep-Learning-Methoden und der Einflüsse der Parameterwahl ist essentiell, um diese Methoden im Bereich medizinischer Bilddatensätze erfolgreich einsetzen zu können und die Vorhersage des Behandlungserfolgs bei Tumorpatienten für die Individualisierung der Strahlentherapie zu verbessern.

6 Literaturverzeichnis

[1] Aerts H.J.W.L., et al.: *Decoding tumor phenotype by noninvasive imaging using a quantitative radiomics approach.* Nature Communications, 5:4006, (2014).

[2] Buzug T.M.: *Computed Tomography, From Photon Statistics to Modern Cone-Beam CT.* Springer-Verlag, Berlin Heidelberg, (2008).

[3] Chollet F., et al.: *Keras.*, Published: `https://keras.io` (2015).

[4] Demleitner M.: *Skripte zur Entropie, Universität Heidelberg.* `http://www.cl.uni-heidelberg.de/kurs/skripte/stat/html/page017.html` `http://www.cl.uni-heidelberg.de/kurs/skripte/stat/html/page018.html` `http://www.cl.uni-heidelberg.de/kurs/skripte/stat/html/page019.html`, Zugriff am: 03.05.2018.

[5] Deng J., et al.: *ImageNet: A large-scale hierarchical image database.* Computer Vision and Pattern Recognition, IEEE Conference, (2009).

[6] Deutsches Krebsforschungszentrum: *Krebsstatistiken.* DKFZ, (2017) `https://www.krebsinformationsdienst.de/grundlagen/krebsstatistiken.php`, Letztes Update: Dezember 2017, Zugriff am 17.04.2018.

[7] Donahue J., et al.: *Decaf: A Deep Convolutional Activation Feature for Generic Visual Recognition.* Proceedings of the 31st International Conference on Machine Learning, 647-655, China, (2014).

[8] Gibson E., et al.: *NiftyNet: a deep-learning platform for medical imaging.* arXiv preprint, arXiv:1709.03485, (2017).

[9] Glorot X., et al.: *Domain Adaptation for Large-Scale Sentiment Classification: A Deep Learning Approach.* Proceedings of the 28th International Conference on Machine Learning, 513-520, USA, (2011).

[10] Harrel F.E., et al.: *Multivariable prognostic models: Issues in developing models, evaluating assumptions and adequacy, and measuring and reducing errors.* Statistics in Medicine, 15:361-387, (1996).

[11] Hastie T., et al.: *The Elements of Statistical Learning; Data Mining, Inference, and Prediction.* Springer Series in Statistics, Springer, 2. Auflage, (2009).

[12] Henniger J.: *Vorlesung Physikalische Vertiefung: Wechselwirkung von Teilchen mit Materie, Sommersemester 2018.* Institut für Kern- und Teilchenphysik, AG Strahlungsphysik, TU Dresden, (2018).

[13] International Agency for Research on Cancer, WHO: *Cancer Fact Sheets: All Cancers Excluding Non-Melanoma Skin Cancer.* International Agency for Research on Cancer, World Health Organization, (2016)
`http://gco.iarc.fr/today/data/pdf/fact-sheets/cancers/cancer-fact-sheets-29.pdf`, Zugriff am: 17.04.2018.

[14] Kieseritzky, K.v.: *Kopf-Hals-Tumoren - Überblick.* Onko Internetportal, Deutsche Krebsgesellschaft, (2017)
`https://www.krebsgesellschaft.de/onko-internetportal/basis-informationen-krebs/krebsarten/andere-krebsarten/kopf-hals-tumoren/definition-und-haeufigkeit.html`, Letztes Update: Dezember 2017, Zugriff am: 15.05.2018.

[15] Kingma D.P., et al.: *Adam: A Method for Stochastic Optimization.*, arXiv preprint, arXiv:1412.6980, (2014).

[16] Krieger H.: *Strahlungsquellen für Technik und Medizin.* Springer Spektrum, Wiesbaden, 2. Auflage, (2013).

[17] Lao J., et al.: *A Deep Learning-Based Radiomics Model for Prediction of Survival in Glioblastoma Multiforme.* Scientific Reports, 7:10353, (2017).

[18] Larobina M., et al.: *Medical Image File Formats.* Journal of Digital Imaging, 27:200-206, (2014).

[19] LeCun Y., et al.: *Deep learning.* Nature, 521(7553):436, (2015).

[20] Leger S., et al.: *A comparative study of machine learning methods for time-to-event survival data for radiomics risk modelling.* Scientific Reports, 7:13206, (2017).

[21] Li Q., et al.: *A Fully-Automatic Multiparametric Radiomics Model: Towards Reproducible and Prognostic Imaging Signature for Prediction of Overall Survival in Glioblastoma Multiforme.* Scientific Reports, 7:14331, (2017).

[22] Li Z., et al.: *Deep Learning based Radiomics (DLR) and its usage in noninvasive IDH1 prediction for low grade glioma.* Scientific Reports, 7:5467, (2017).

[23] Linge A., et al.: *HPV status, cancer stem cell marker expression, hypoxia gene signatures and tumour volume identify good prognosis subgroups in patients with HNSCC after primary radiochemotherapy: A multicentre retrospective study of the German Cancer Consortium Radiation Oncology Group (DKTK-ROG).* Radiotherapy and Oncology, 121(3):364–373, (2016).

[24] Löck S., et al.: *Residual tumour hypoxia in head-and-neck cancer patients undergoing primary radiochemotherapy, final results of a prospective trial on repeat FMISO-PET imaging.* Radiotherapy and Oncology, 124(3):533-540, (2017).

[25] Maurer H.-J., Zieler E. (Hrsg.): *Physik der Bildgebenden Verfahren in der Medizin.* Springer-Verlag, Berlin Heidelberg New York Tokyo, (1984).

[26] Mildenberger P., et al.: *Introduction of the DICOM standard.*, European Radiology, 12:920-927, (2002).

[27] Nie D., et al.: *3D Deep Learning for Multi-modal Imaging-Guided Survival Time Prediction of Brain Tumor Patients.* Medical Image Computing and Computer-Assisted Intervention – MICCAI 2016, Springer, Cham, 9901:212-220, (2016).

[28] Nielsen M.A.: *Neural Networks and Deep Learning.* Determination Press, (2015) `http://neuralnetworksanddeeplearning.com/`, Letztes Update: Dezember 2017, Zugriff am 23.04.2018.

[29] Paul R., et al.: *Deep Feature Transfer Learning in Combination with Traditional Features Predicts Survival Among Patients with Lung Adenocarcinoma.* Tomography: a journal for imaging research, 2(4):388, (2016).

[30] Simonyan K., et al.: *Very Deep Convolutional Networks for Large-Scale Image Recognition.* arXiv preprint, arXiv:1409.1556, (2015).

[31] Spearman C.: *Correlation calculated from faulty data.* British Journal of Psychology, 3:271–295, (1910).

[32] Srivastava N., et. al.: *Dropout: A Simple Way to Prevent Neural Networks from Overfitting.* The Journal of Machine Learning Research, 15(1):1929-1958, (2014).

[33] Stolz W.: *Radioaktivität, Grundlagen - Messung - Anwendung.* B. G. Teubner Verlag, Wiesbaden, 5. Auflage, (2005).

[34] Theano Development Team: *Theano: A Python Framework for fast computation of mathematical expressions.* arXiv preprint, arXiv:1605.02688, (2016).

[35] Wegener O.H.: *Grundkurs Computertomographie.* Blackwell Wissenschafts-Verlag, Berlin Wien, (1996).

[36] Würschig T.: *Aufbau eines Versuchsplatzes für die Positronen-Emissions-Tomographie.* (Diplomarbeit), TU Dresden, (2005).

[37] Yosinski J., et al.: *How transferable are features in deep neural networks?* Advances in Neural Information Processing Systems, 3320-3328, (2014).

[38] Zips D., et al.: *Exploratory prospective trial of hypoxia-specific PET imaging during radiochemotherapy in patients with locally advanced head-and-neck cancer.* Radiotherapy and Oncology 105(1):21–28, (2012).

[39] (Unbekannt): *Receiver Operating Characteristic (ROC) Curve: Definition, Example.* (2016)
`http://www.statisticshowto.com/receiver-operating-characteristic-roc-curve/`,
Letztes Update: Oktober 2017, Zugriff am: 04.05.2018.

A Anhang

A.1 CT-Generationen

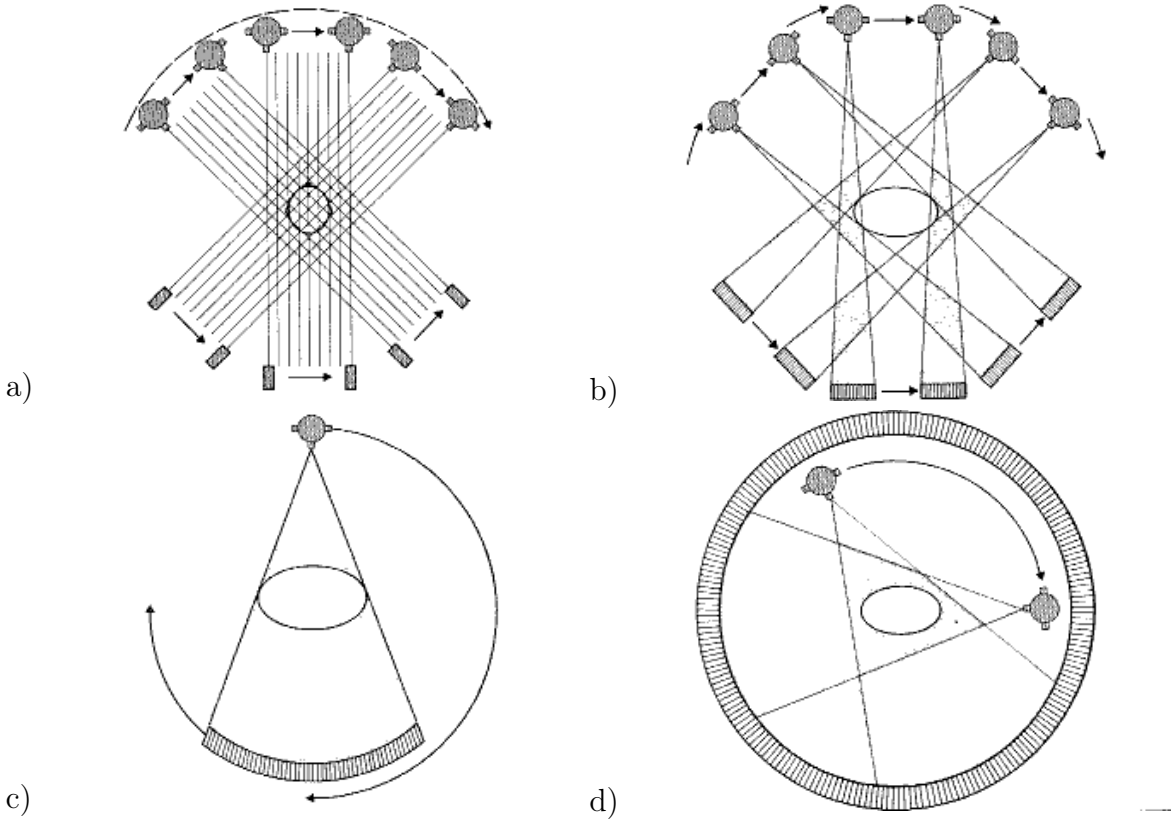


Abbildung A.1: Aufnahmeprinzipien der verschiedenen CT-Generationen. a) 1. CT-Generation: Einzel-Detektor-Rotations-Translations-Scanner. b) 2. CT-Generation: Mehr-Detektor-Rotations-Translations-Scanner. c) 3. CT-Generation: Rotationsscanner mit beweglichem Detektorsystem. d) 4. CT-Generation: Rotationssystem mit stationären Detektoren. [35]

A.2 Fourier-Scheiben-Theorem

Herleitung des FOURIER-Scheiben-Theorems [2]:

$$\tilde{P}_\gamma(q) = \mathcal{F}_1(p_\gamma(\xi)) = \int_{-\infty}^{\infty} p_\gamma(\xi)\exp(-2\pi \mathrm{i} q\xi)\,\mathrm{d}\xi \tag{A.1}$$

$$= \int_{-\infty}^{\infty}\left(\int_{-\infty}^{\infty}\mu(\xi,\eta)\,\mathrm{d}\eta\right)\exp(-2\pi \mathrm{i} q\xi)\,\mathrm{d}\xi \quad | \text{ Projektion} \tag{A.2}$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\mu(\xi,\eta)\exp(-2\pi \mathrm{i} q\xi)\,\mathrm{d}\eta\mathrm{d}\xi \tag{A.3}$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty}\mu(\xi(x,y),\eta(x,y))\exp(-2\pi \mathrm{i} q\xi(x,y))\,\mathrm{d}x\mathrm{d}y \quad | \text{ Rotation} \tag{A.4}$$

$$= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\exp(-2\pi \mathrm{i}(qx\cos\gamma + qy\sin\gamma))\,\mathrm{d}x\mathrm{d}y \tag{A.5}$$

$$\overset{\substack{u=q\cos\gamma\\ v=q\sin\gamma}}{=} \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(x,y)\exp(-2\pi \mathrm{i}(ux+vy)\,\mathrm{d}x\mathrm{d}y \tag{A.6}$$

$$= \mathcal{F}_2(f(x,y)) = \tilde{F}(u,v)|_{u=q\cos\gamma,\, v=q\sin\gamma} \tag{A.7}$$

$$\square \tag{A.8}$$

A.3 Funktionsweise der Schichten im CNN

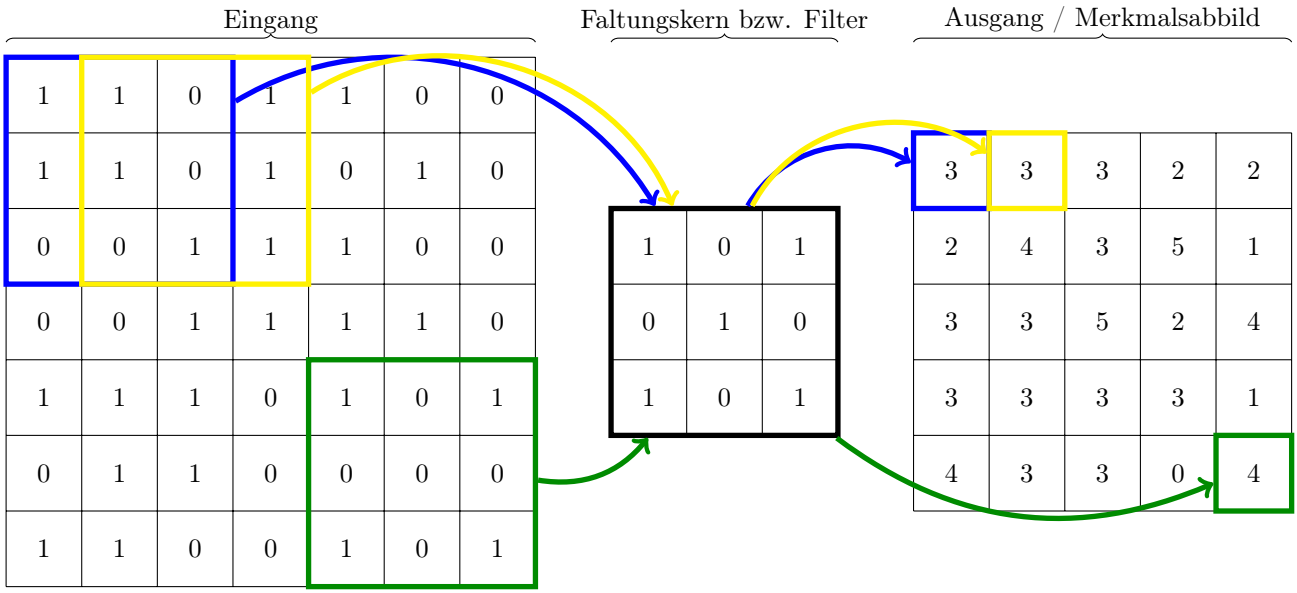


Abbildung A.2: Funktionsweise einer Conv-Schicht anhand eines Beispiels. Der Faltungskern bzw. Filter wird elementweise mit einer Region multipliziert und die (in diesem Fall neun) Ergebnisse aufsummiert. Das Ergebnis ist ein Eintrag in dem zum Filter gehörenden Merkmalsabbild. Die Zuordnung Region-Eintrag ist hier anhand von drei Beispielen dargestellt.

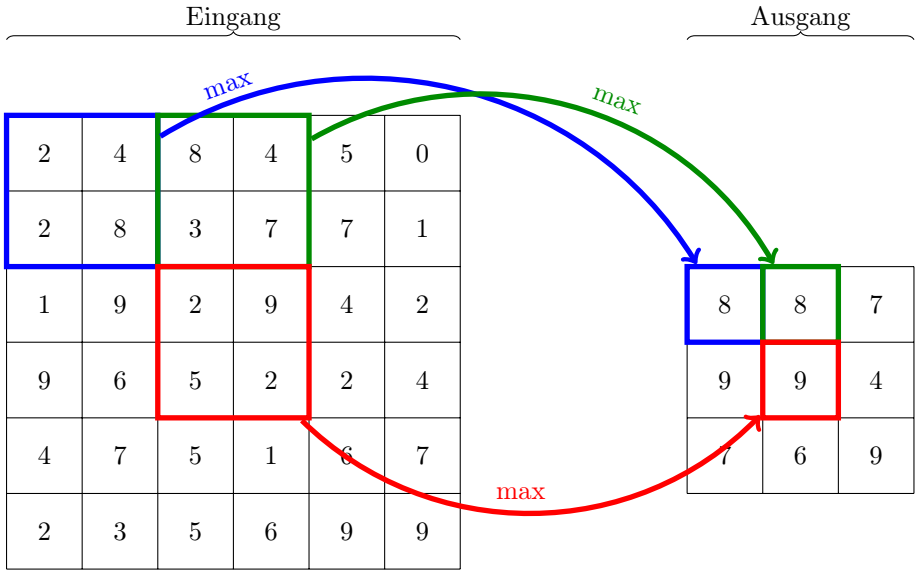


Abbildung A.3: Funktionsweise einer Pool-Schicht anhand eines Beispiels. Bei diesem Beispiel wird Maxpooling (das Maximum der gewählten Region wird verwendet) mit einer Kernelgröße von 2×2 angewendet. Dadurch reduziert sich die Dimension des Eingangs in jeder Richtung auf die Hälfte.

A.4 Flussdiagramm Binarisierung

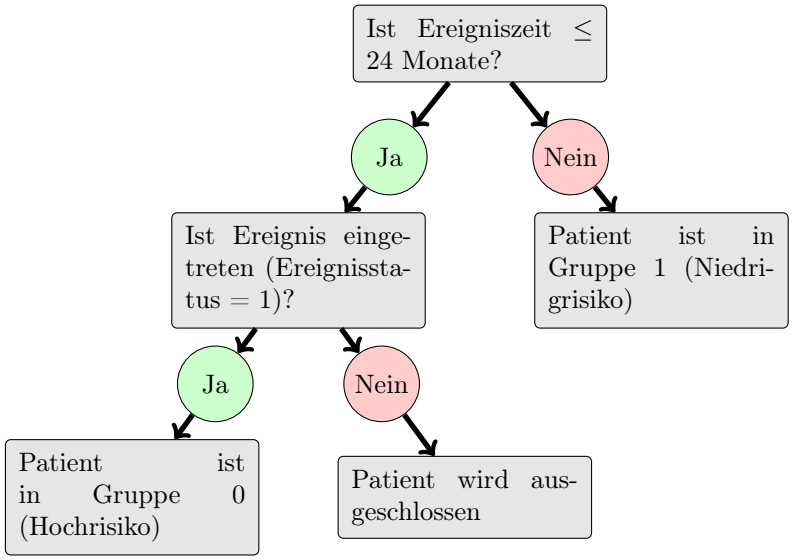


Abbildung A.4: Flussdiagramm zur Binarisierung. Stellt den Ablauf des Binarisierungsvorgangs für ein allgemeines Ereignis und der Zeitgrenze von 24 Monaten dar.

A.5 Aufbau des selbst trainierten Netzes

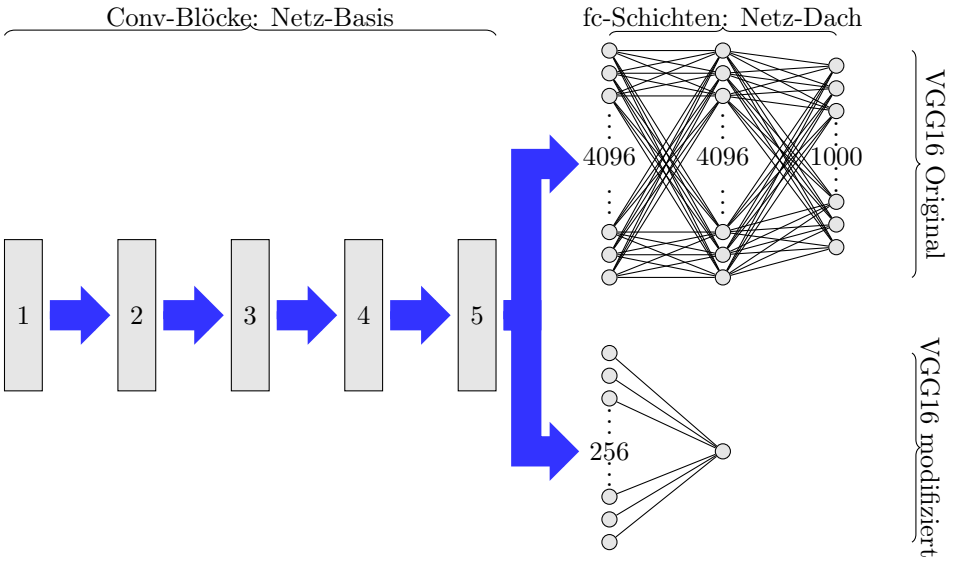


Abbildung A.5: Schematischer Aufbau des modifizierten VGG16 im Vergleich zum Originalnetz. In der modifizierten Variante wurden die letzten beiden Schichten selbst trainiert. Die Conv-Blöcke sind dieselben, die im originalen VGG16 verwendet werden und schon vortrainiert. Zu beachten ist, dass die Neuronen der ersten fc-Schicht auch schon mit allen Einheiten der letzten Conv-Schicht voll vernetzt sind (hier nicht dargestellt). Genauere Informationen zum Aufbau und zu den verwendeten Parametern im originalen VGG16 sind in [30] zu finden.

A.6 Grauwert-Reskalierung

Seien die Grauwerte vor der Reskalierung im Intervall $[min_{\text{alt}}, max_{\text{alt}}]$ und sollen anschließend im Intervall $[min_{\text{neu}}, max_{\text{neu}}]$ sein, sodass die Information bezüglich des Maximums/Minimums erhalten bleibt. Dann muss gelten:

$$\frac{x_{\text{neu}} - max_{\text{neu}}}{x_{\text{neu}} - min_{\text{neu}}} = \frac{x_{\text{alt}} - max_{\text{alt}}}{x_{\text{alt}} - min_{\text{alt}}} \tag{A.9}$$

$$x_{\text{neu}} - max_{\text{neu}} = \frac{x_{\text{alt}} - max_{\text{alt}}}{x_{\text{alt}} - min_{\text{alt}}} \cdot (x_{\text{neu}} - min_{\text{neu}}) \tag{A.10}$$

$$x_{\text{neu}} = \frac{max_{\text{neu}} - min_{\text{neu}} \cdot \frac{x_{\text{alt}} - max_{\text{alt}}}{x_{\text{alt}} - min_{\text{alt}}}}{1 - \frac{x_{\text{alt}} - max_{\text{alt}}}{x_{\text{alt}} - min_{\text{alt}}}} \tag{A.11}$$

$$x_{\text{neu}} = \frac{x_{\text{alt}}(max_{\text{neu}} - min_{\text{neu}}) + max_{\text{alt}} \cdot min_{\text{neu}} - max_{\text{neu}} \cdot min_{\text{alt}}}{max_{\text{alt}} - min_{\text{alt}}} \tag{A.12}$$

A.7 Schematischer Arbeitsablauf des Radiomics-Grundgerüsts

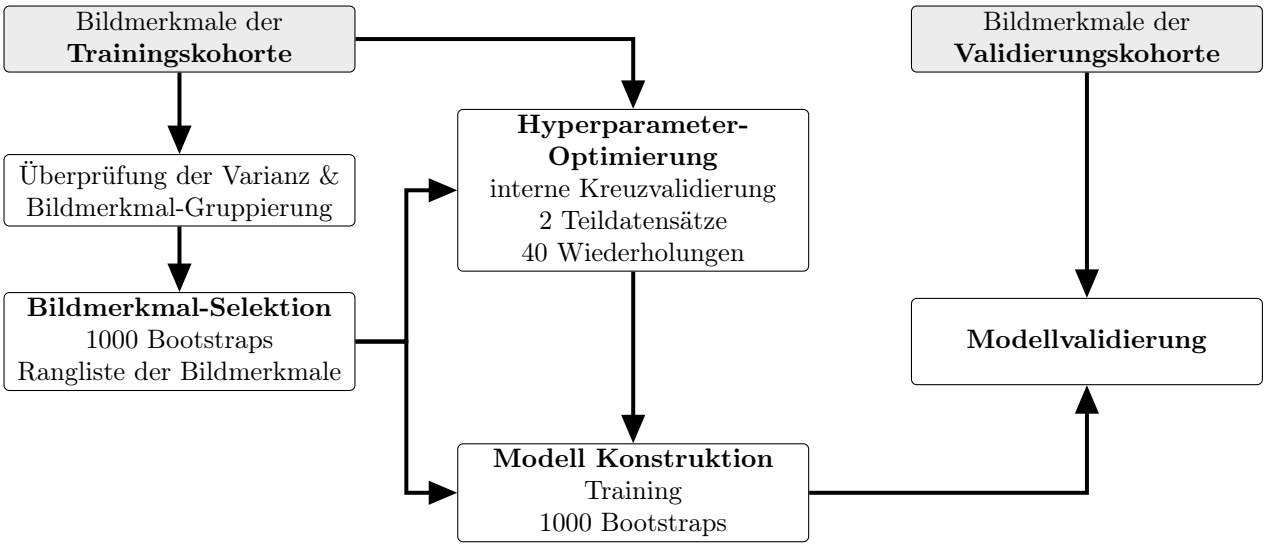


Abbildung A.6: Modellkonstruktion mit dem Radiomics-Grundgerüst als schematisches Flussdiagramm. Grafik orientiert an [20].

A.8 Bildmerkmal-Selektions- und Machine-Learning-Methoden

In diesem Abschnitt wird die betrachtete Zufallsvariable mit $X(Y,T)$, der n-dimensionale Bildmerkmal-Vektor mit $Y = (y_1, \ldots, y_n)$ und die richtige Risikogruppe (Therapieausgang) mit T bezeichnet [20].

A.8.1 Bildmerkmal-Selektions-Methoden

- SPEARMAN-Korrelation (Spearman):

Der SPEARMAN-Korrelationskoeffizient r_s ist ein Maß für die parameterfreie Korrelation zweier Variablen (vgl. [20]). Für ein Bildmerkmal $y \in Y$ und den dazugehörigen Therapieausgang T ist er gegeben durch:

$$r_s = \frac{\sum_{i=1}^{l}(\mathrm{rg}(y_i) - \overline{\mathrm{rg}_y})(\mathrm{rg}(T_i) - \overline{\mathrm{rg}_T})}{\sqrt{\sum_{i=1}^{l}(\mathrm{rg}(y_i) - \overline{\mathrm{rg}_y})^2}\sqrt{\sum_{i=1}^{l}(\mathrm{rg}(T_i) - \overline{\mathrm{rg}_T})^2}} \ . \tag{A.13}$$

Dabei wird jeweils über alle l Patienten summiert und rg() steht für den Rang (die Position in der Reihe der Patienten, wenn die Werte der Größe nach geordnet wurden) des Bildmerkmals bzw. des Risikogruppenwertes [31].

- Mutual Information Maximisation (MIM):

Diese Methode schätzt die Relevanz des Bildmerkmals $y \in Y$ für den damit verbundenen Therapieausgang T ab. Dafür wird eine lineare Anpassung verwendet, die auf der Korrelation $\rho(y,T) = 2 \cdot (\text{C-Index}^1 - 0{,}5)$ basiert. Es ergibt sich die Mutual Information I durch (vgl. [20]):

$$I = -0{,}5 \cdot \ln(1 - \rho(y,T)^2) \ . \tag{A.14}$$

- Mutual Information Feature Selection MIFS):

Dieser Algorithmus sucht und selektiert eine Teilmenge $S \subset Y$ der Bildmerkmale, welche die Zielfunktion Ω maximiert; unter Verwendung von Gleichung (A.14):

$$\Omega = \underset{y \in Y}{\mathrm{argmax}}\, I(y,T) + \beta \sum_{s_j \in S} I(y, s_j) \ . \tag{A.15}$$

Dabei wurde der Wert $\beta = 1$ verwendet (vgl. [20]).

- Minimum Redundancy Maximum Relevance (MRMR):

Diese Methode kombiniert die maximale Mutual Information zwischen den Bildmerkmalen in der Teilmenge $S \subset Y$ und dem Therapieausgang T mit der Bedingung, dass die Bildmerkmale aus S minimal redundant sein sollen. Dafür werden mit einer schrittweisen Suche, die auf der

[1] Concordance Index [10, 31]

Mutual Information I basiert, die Bildmerkmale gesucht, welche Ω maximieren (vgl. [20]):

$$\Omega = \underset{y \in Y \setminus S}{\operatorname{argmax}} I(y, T) - \frac{1}{|S|} \sum_{s_j \in S} I(y, s_j) \ . \tag{A.16}$$

- Random Forest Variable Importance (RFVI):

Dieser Algorithmus benutzt Trainingsdaten aus k Bootstrap-Stichproben, um einen Wald (Forest) aus k Entscheidungsbäumen mit m zufällig ausgewählten Bildmerkmalen anzupassen. Diese werden aufsteigend nach ihrer minimalen „Tiefe“ (relativer Abstand zum Ausgangsknoten eines Entscheidungsbaumes über alle Bäume des Forest betrachtet) angeordnet und fortlaufend hinzu genommen, bis die gemeinschaftliche Variablenwichtigkeit nicht mehr zunimmt. Diese wird berechnet, indem die Bildmerkmale permutiert und anschließend die Veränderung im Vorhersagefehler zwischen vorherigem und neuem Forest berechnet werden. Dieser Prozess wird N mal wiederholt und die Bildmerkmale werden mithilfe ihrer durchschnittlichen minimalen Tiefe in eine Rangliste eingeordnet (vgl. [20]).

A.8.2 Klassifikationsmethoden

Alle Methoden zielen darauf ab, eine optimale, abgrenzende Hyperfläche im Bildmerkmal-Raum zu finden, wodurch die Bildmerkmale für die Klassifikation unterteilt werden [11].

- Logistische Regression (LogReg):

Die logistische Regression ist eine lineare Klassifikationsmethode. Es wird mit den Bildmerkmalen Y die Wahrscheinlichkeit $p_X(Y, \beta_{10}, \beta_1)$ für die Klasse $X \in \{0, 1\}$ vorhergesagt. Diese ist gegeben durch [11]:

$$p_0(Y, \beta_{10}, \beta_1) = \frac{\exp(\beta_{10} + \beta_1^T Y)}{1 + \exp(\beta_{10} + \beta_1^T Y)} \ , \tag{A.17}$$

$$p_1(Y, \beta_{10}, \beta_1) = \frac{1}{1 + \exp(\beta_{10} + \beta_1^T Y)} = 1 - p_0(Y, \beta_{10}, \beta_1) \ . \tag{A.18}$$

Ziel ist es nun, die am besten passenden Parameter β_{10} und β_1 zu finden, mit denen möglichst viele Patienten gut den Risikogruppen 0 oder 1 zugeordnet werden können [11].

- Naive Bayes Klassifizierer (naiveBayes):

Dieses Modell beruht auf der Schätzung von Wahrscheinlichkeitsdichten und verwendet zur Berechnung einer Klassenwahrscheinlichkeit das Bayes-Theorem. Für jede Klasse j wird eine Dichte f_j und eine A-priori-Wahrscheinlichkeit π_j geschätzt. Es wird davon ausgegangen, dass für eine gegebene Klasse j die Bildmerkmale $y_k \in Y$ unabhängig sind [11]:

$$f_j(Y) = \prod_{k=1}^{n} f_{jk}(y_k) \ . \tag{A.19}$$

Die individuellen klassenabhängigen Marginalverteilungen f_{jk} können dann geschätzt werden,

wodurch sich die folgendenden Wahrscheinlichkeiten ergeben:

$$p_0(Y) = \frac{\exp(\alpha_0 + \sum_{k=1}^{n} g_{0k}(y_k))}{1 + \exp(\alpha_0 + \sum_{k=1}^{n} g_{0k}(y_k))} , \tag{A.20}$$

$$p_1(Y) = \frac{1}{1 + \exp(\alpha_0 + \sum_{k=1}^{n} g_{0k}(y_k))} = 1 - p_0(Y) . \tag{A.21}$$

Die Unbekannten α_j und $g_{jk}(y_k)$ werden dann an das Problem angepasst [11].

- Support Vector Machines (SVM):

Diese Methoden produzieren nichtlineare Klassifikationsgrenzen, die darauf abzielen, Klassen zu seperieren, die durch lineare Grenzen nicht seperiert werden können. Die abgrenzende Hyperfläche wird durch einen Vektor β und dem Parameter β_0 definiert. Für die Bildmerkmale Y_i und die dazugehörigen Zielwerte z_i ergibt sich dann die Minimierungsaufgabe:

$$\min ||\beta|| \quad \text{bezogen auf} \begin{cases} z_i(Y_i^T \beta + \beta_0) \geq 1 - \xi_i \ \ \forall i, \\ \xi_i \geq 0, \ \sum \xi_i \leq \text{Konstante} . \end{cases} \tag{A.22}$$

Dadurch wird die breiteste Grenze im Raum der Bildmerkmale gefunden, durch welche die Bildmerkmale nach den Klassen unterteilt werden [11].

- Random Forest (RF):

Auf Grundlage von Bootstraps der Trainigsdaten werden zufällige Entscheidungsbäume erstellt. Dazu werden an jedem Knoten zufällig m Variablen aus den n Bildmerkmalen gewählt, der beste Trennungspunkt auf Grundlage der m Variablen und der richtigen Klasse bestimmt und mit diesem Trennungspunkt der Knoten in zwei Tochterknoten geteilt. Die „gewachsenen“ Bäume bilden dann den Klassifizierer. Die Klassifikation der Daten findet dann statt, indem die Klasse gewählt wird, die von den meisten Bäumen aus dem Forest vorhergesagt wurde [11].

A.9 Verläufe der Netzwerktrainings

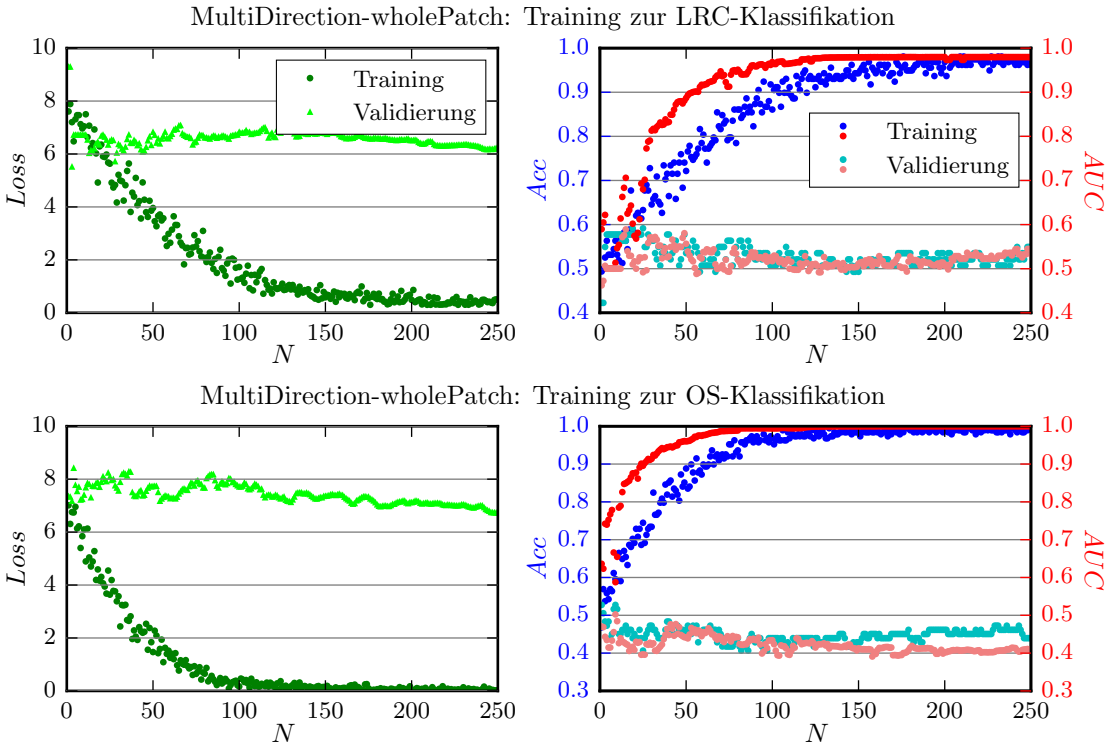


Abbildung A.7: „MultiDirection-wholePatch"-Datensatz (Training und Validierung) für loko-regionäre Tumorkontrolle (LRC, oben) und Gesamtüberleben (unten). Gezeigt ist die Loss-Funktion (links) sowie die Genauigkeit *Acc* und die *AUC* (rechts) nach jeder Trainingsepoche N.

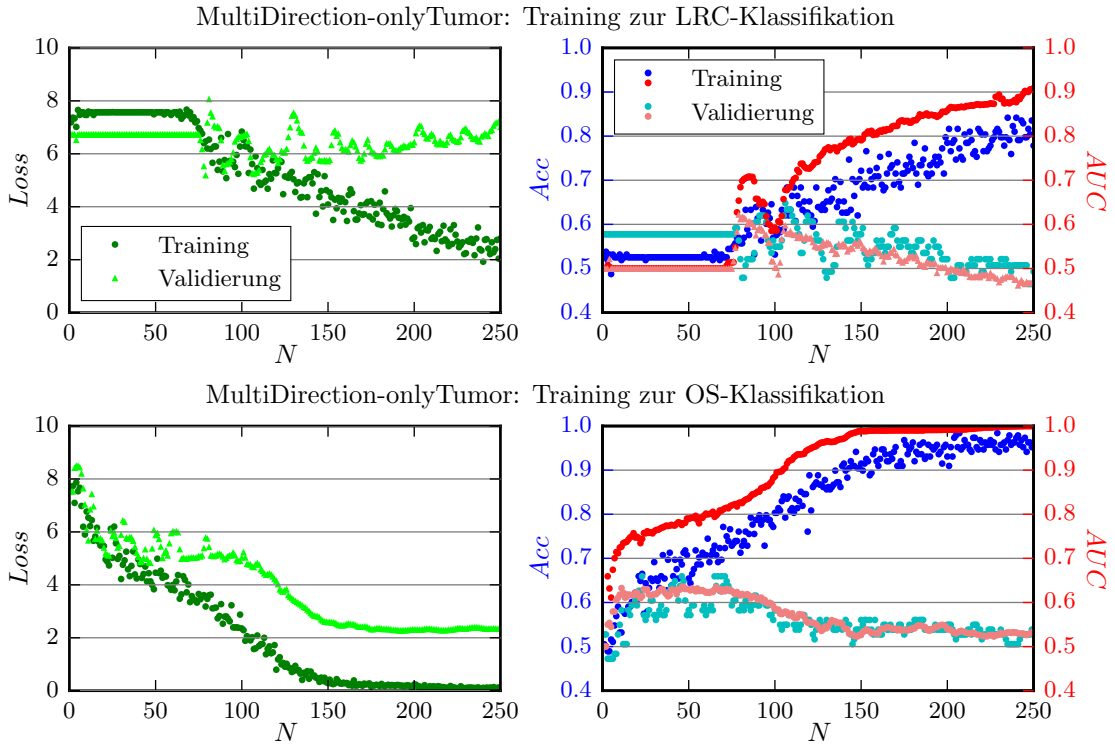


Abbildung A.8: „MultiDirection-onlyTumor“-Datensatz (Training und Validierung) für loko-regionäre Tumorkontrolle (LRC, oben) und Gesamtüberleben (unten). Gezeigt ist die Loss-Funktion (links) sowie die Genauigkeit *Acc* und die *AUC* (rechts) nach jeder Trainingsepoche N.

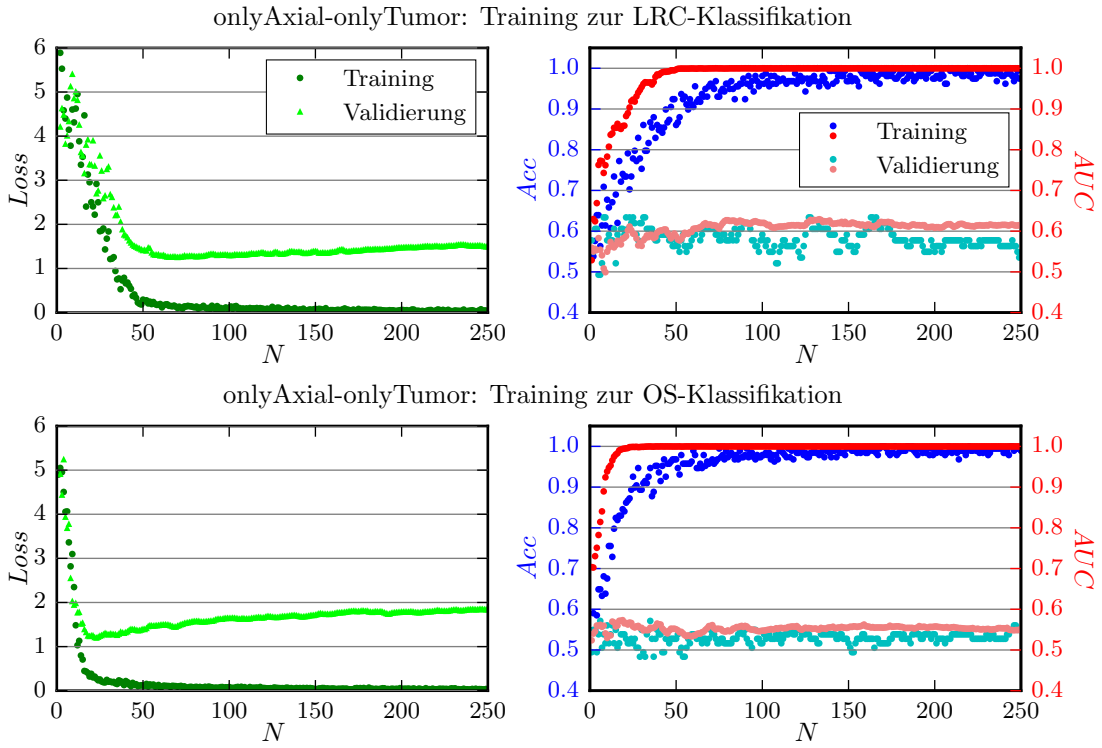


Abbildung A.9: „onlyAxial-onlyTumor"-Datensatz (Training und Validierung) für lokoregionäre Tumorkontrolle (LRC, oben) und Gesamtüberleben (unten). Gezeigt ist die Loss-Funktion (links) sowie die Genauigkeit *Acc* und die *AUC* (rechts) nach jeder Trainingsepoche *N*.

A.10 Ergebnisse der Radiomics-Modellierung

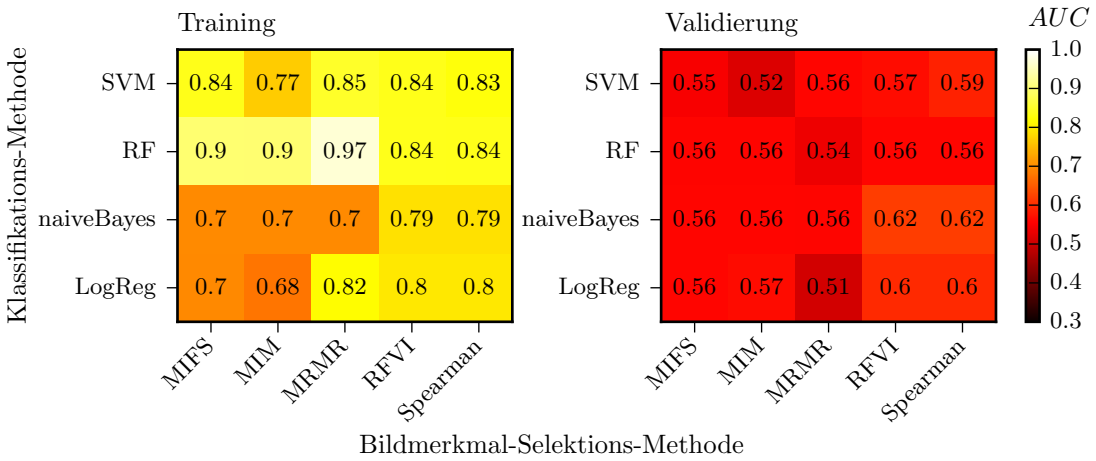


Abbildung A.10: Klassifizierungsleistungen der verschiedenen Modelle nach dem maschinellen Lernen zur Modelloptimierung (LRC-Risikovorhersage mit TL-Deep-Bildmerkmalen aus „onlyAxial-wholePatch“-Datensatz)

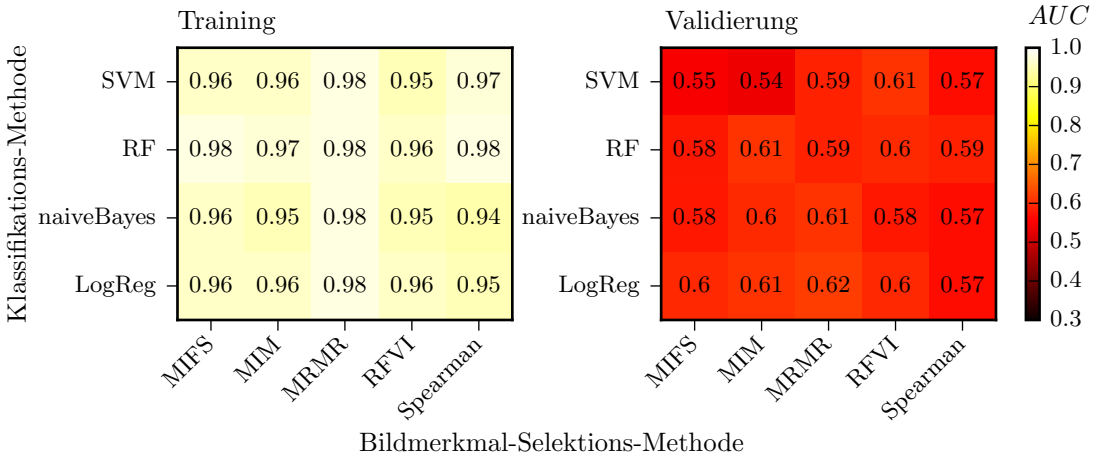


Abbildung A.11: Klassifizierungsleistungen der verschiedenen Modelle nach dem maschinellen Lernen zur Modelloptimierung (LRC-Risikovorhersage mit ST-Deep-Bildmerkmalen aus „onlyAxial-onlyTumor“-Datensatz).

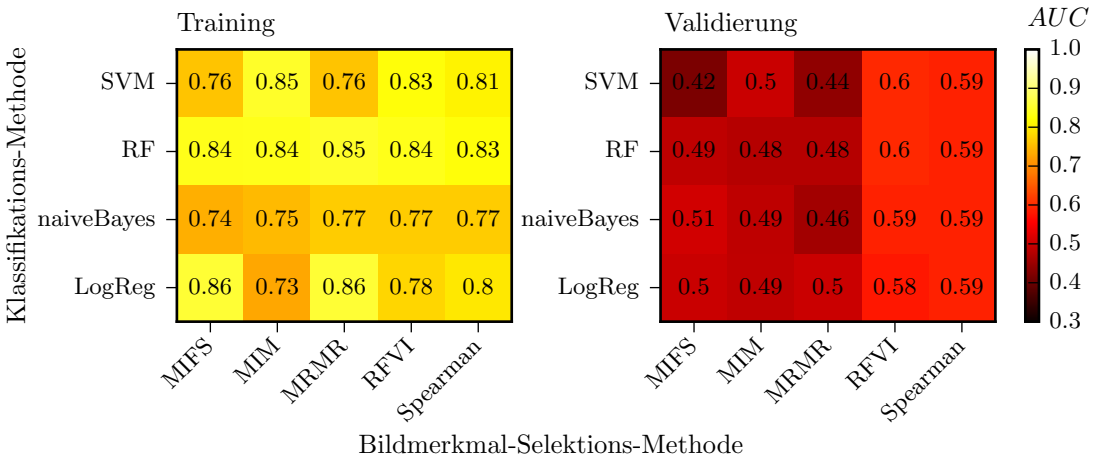


Abbildung A.12: Klassifizierungsleistungen der verschiedenen Modelle nach dem maschinellen Lernen zur Modelloptimierung (LRC-Risikovorhersage mit TL-Deep-Bildmerkmalen aus „onlyAxial-onlyTumor“-Datensatz).

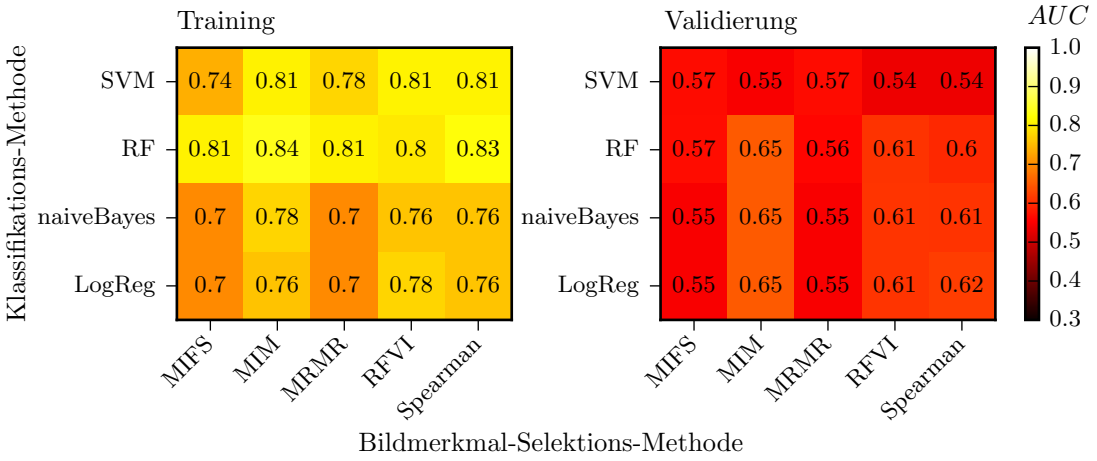


Abbildung A.13: Klassifizierungsleistungen der verschiedenen Modelle nach dem maschinellen Lernen zur Modelloptimierung (LRC-Risikovorhersage mit konventionellen Radiomics-Bildmerkmalen).

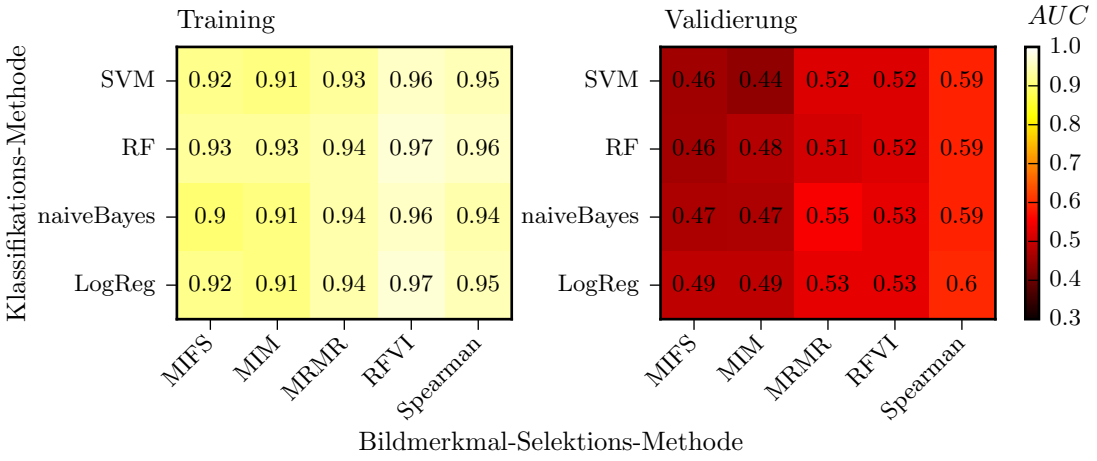


Abbildung A.14: Klassifizierungsleistungen der verschiedenen Modelle nach dem maschinellen Lernen zur Modelloptimierung (OS-Risikovorhersage mit ST-Deep-Bildmerkmalen aus „onlyAxial-wholePatch“-Datensatz).

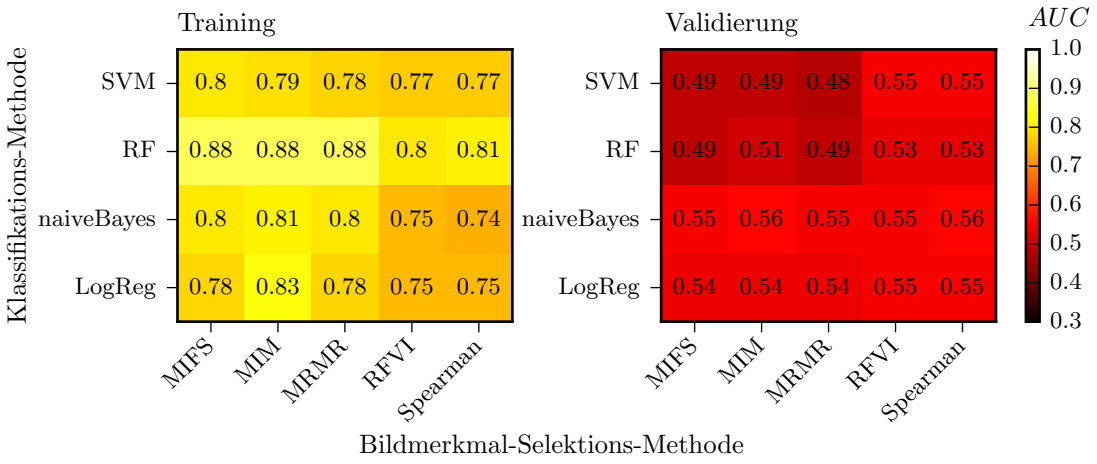


Abbildung A.15: Klassifizierungsleistungen der verschiedenen Modelle nach dem maschinellen Lernen zur Modelloptimierung (OS-Risikovorhersage mit TL-Deep-Bildmerkmalen aus „onlyAxial-wholePatch“-Datensatz).

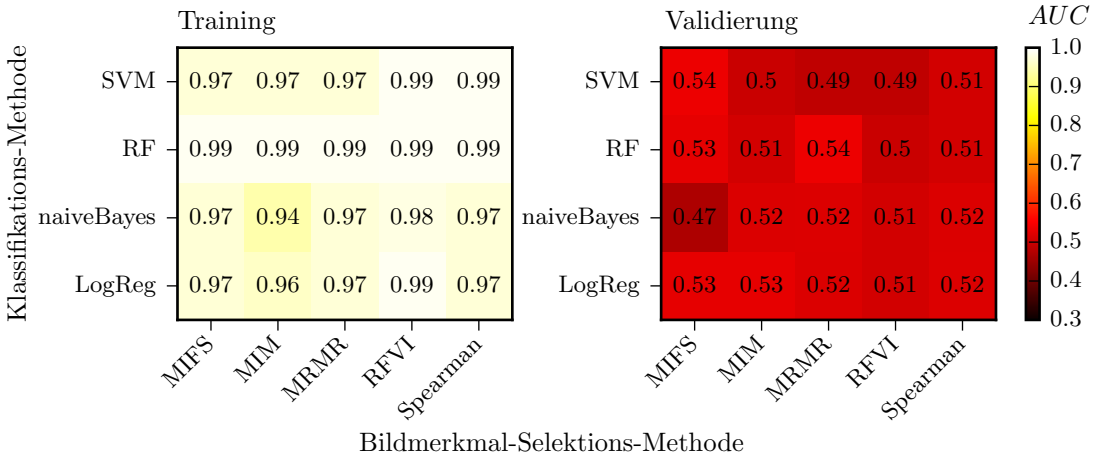


Abbildung A.16: Klassifizierungsleistungen der verschiedenen Modelle nach dem maschinellen Lernen zur Modelloptimierung (OS-Risikovorhersage mit ST-Deep-Bildmerkmalen aus „onlyAxial-onlyTumor“-Datensatz).

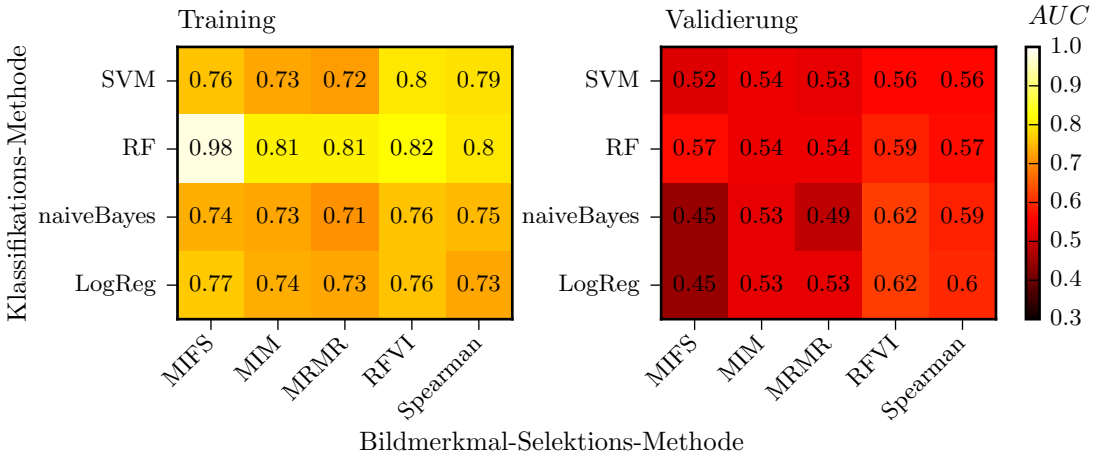


Abbildung A.17: Klassifizierungsleistungen der verschiedenen Modelle nach dem maschinellen Lernen zur Modelloptimierung (OS-Risikovorhersage mit TL-Deep-Bildmerkmalen aus „onlyAxial-onlyTumor“-Datensatz).

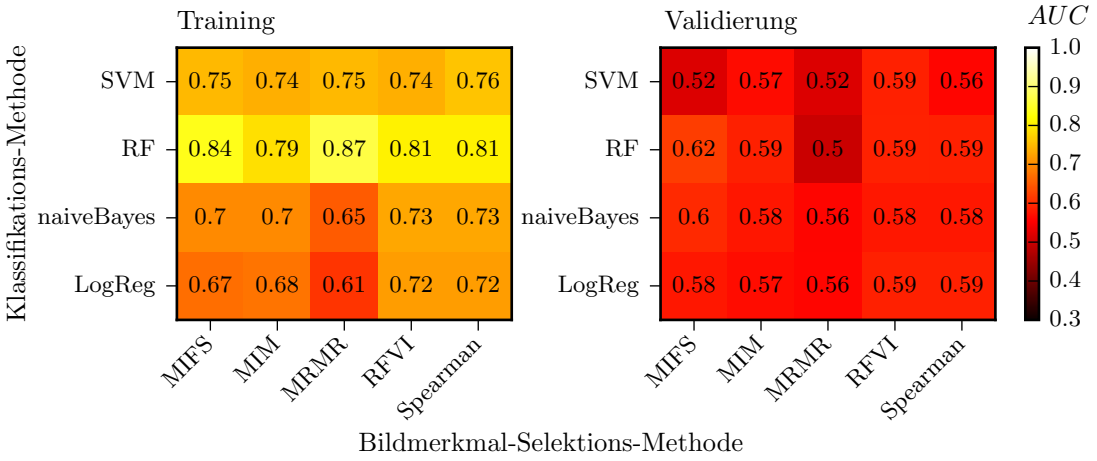


Abbildung A.18: Klassifizierungsleistungen der verschiedenen Modelle nach dem maschinellen Lernen zur Modelloptimierung (OS-Risikovorhersage mit konventionellen Radiomics-Bildmerkmalen).

A.11 ROC-Kurven zu Schwellenwertbestimmung der Kaplan-Meier-Analysen

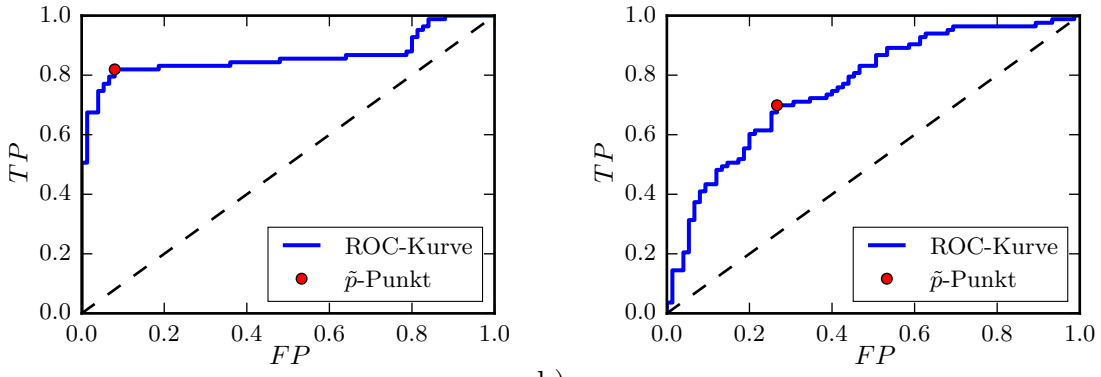


Abbildung A.19: a) ROC-Kurve für die Klassifikation der Patienten mit dem MRMR-LogReg-Modell, das mit TL-Deep-onlyTumor-Bildmerkmalen für LRC trainiert wurde. b) ROC-Kurve für die Klassifikation der Patienten mit dem MIM-LogReg-Modell, das mit konventionellen Radiomics-Bildmerkmalen für LRC trainiert wurde. Der $\tilde{p}$-Punkt ist am weitesten von der Diagonalen entfernt.

Danksagung

Ich möchte mich hiermit bei allen Personen bedanken, die mich während der Arbeit oder im Vorfeld unterstützt haben. Einen ganz besonderen Dank möchte ich folgenden Personen übermitteln:
Als erstes möchte ich mich bei Herrn Prof. Dr. Arno Straessner bedanken, der sich bereit erklärt hat, für die Arbeit als Erstgutachter zu fungieren.
Des Weiteren möchte ich mich herzlichst bei Herrn PD Dr. Steffen Löck bedanken, der zum einen die Rolle des Zweitgutachters übernommen hat und zum anderen als Gruppenleiter der Forschungsgruppe für Modellierung und Biostatistik in der Radioonkologie mir überhaupt erst ermöglicht hat, diese Bachelorarbeit am OncoRay zu verfassen. Außerdem hat er mich mit betreut und war immer für alle Fragen offen.
Für alle Fragen offen waren auch die anderen Mitarbeiter der Forschungsgruppe. Deshalb gilt mein Dank auch all jenen. Allen voran möchte ich davon Herrn Stefan Leger und Herrn Dr. Alex Zwanenburg-Bezemer danken, die meine Betreuung zusammen mit Steffen Löck hauptsächlich übernommen haben und mir, wann immer nötig, helfend und unterstützend zur Seite standen.
Schließlich möchte ich noch meinen Freunden und Kommilitonen Jochen, Tom und ganz besonders Carsten für die gegenseitige Motivation und Hilfe während des bisherigen Studiums danken. Herzlich bedanken möchte ich mich ebenfalls bei meiner Freundin Eva, die in der Bearbeitungszeit oft verzichten oder einlenken musste und dennoch immer hinter mir stand, mich motiviert und unterstützt hat. Ebenfalls gilt mein Dank ihren Eltern Katja und Jens, die mich in der Zeit des Studiums immer herzlich aufgenommen haben.
Zuletzt möchte ich mich bei meiner Familie bedanken: Meinen Brüdern für die Motivation sowie meinen Eltern für die gesamte Unterstützung und Hilfe im Studium. Außerdem danke ich ihnen dafür, dass sie mir dieses Studium überhaupt erst ermöglichen.

Erklärung

Hiermit erkläre ich, dass ich diese Arbeit mit dem Titel *Vorhersage des Behandlungserfolgs mittels Deep-Learning-Verfahren zur Individualisierung der Strahlentherapie* im Rahmen der Betreuung am Institut für Kern- und Teilchenphysik und am OncoRay ohne unzulässige Hilfe Dritter verfasst und alle Quellen als solche gekennzeichnet habe.

Leopold Grabs
Dresden, Mai 2018