

Portfoliooptimierung mithilfe von Maschine-Learning Algorithmen auf der Grundlage des Markowitz-Modells

Umsetzung

SAMUEL HESSBERGER

27. Mai 2024

Zusammenfassung

In diesem zweiten Teil der Arbeit wird die praktische Umsetzung und Evaluation von Portfoliooptimierungsmodellen behandelt, die auf dem Mean-Variance-Modell von Harry Markowitz und modernen maschinellen Lernalgorithmen basieren. Aufbauend auf den theoretischen Grundlagen und Methoden des ersten Teils, fokussiert sich diese Studie auf die Implementierung und Leistungsbewertung der entwickelten Modelle.

Die Untersuchung umfasst die Implementierung sowohl traditioneller statistischer Methoden als auch maschineller Lerntechniken, um Vorhersagen zu Aktienkursen zu treffen und optimierte Portfolios zu erstellen. Durch die Prognose zweier Zeiträume, 2019 und Januar bis April 2024, werden die Modelle auf ihre Effizienz und Genauigkeit geprüft. Die Ergebnisse zeigen, dass traditionelle statistische Modelle in bestimmten Szenarien höhere Renditen erzielen können als maschinelle Lernmethoden, obwohl letztere stabilere und konsistente Ergebnisse liefern.

Die Evaluation umfasst einen Vergleich der Performance verschiedener Modelle in unterschiedlichen Zeitperioden und untersucht die Einflüsse von Eingabeparametern auf die Modellgüte. Es wird festgestellt, dass die Qualität und Quantität der Daten sowie die Modellparameter entscheidende Rollen bei den maschinellen Lerntechniken spielen. Weiterhin werden die Herausforderungen und Limitationen der implementierten Modelle diskutiert, insbesondere im Hinblick auf Overfitting und die Praktikabilität in der realen Anwendung.

Diese Arbeit liefert wertvolle Erkenntnisse zu maschinellem Lernen und traditioneller Finanztheorie und zeigt auf, wie diese Ansätze zur Verbesserung der Portfoliooptimierung eingesetzt werden können. Die Ergebnisse bieten eine fundierte Basis für zukünftige Forschungen und Anwendungen in der Finanzmarktanalyse und -prognose.

Abstract

In this second part of the thesis, the practical implementation and evaluation of portfolio optimization models based on Harry Markowitz's mean-variance model and modern machine learning algorithms are discussed. Building on the theoretical foundations and methods of the first part, this study focuses on the implementation and performance evaluation of the developed models.

The investigation includes the implementation of both traditional statistical methods and machine learning techniques to predict stock prices and create optimized portfolios. By forecasting two periods, 2019 and January to April 2024, the models are tested for their efficiency and accuracy. The results show that traditional statistical models can achieve higher returns in certain scenarios compared to machine learning methods, although the latter provide more stable and consistent results.

The evaluation includes a comparison of the performance of different models in various time periods and examines the impact of input parameters on model quality. It is found that the quality and quantity of data, as well as model parameters, play crucial roles in machine learning techniques. Furthermore, the challenges and limitations of the implemented models are discussed, particularly with regard to overfitting and practicality in real-world applications.

This work provides valuable insights into machine learning and traditional financial theory, demonstrating how these approaches can be used to improve portfolio optimization. The findings offer a solid foundation for future research and applications in financial market analysis and forecasting.

Maschinelles Lernen, Mean-Variance-Modell, Markowitz-Modell, LSTM, DAX, Aktienvorhersage, Portfoliooptimierung

INHALTSVERZEICHNIS

I Einführung	3
II Entwurf	3
i Fundamentalanalyse	4
ii Statistische Portfolios	4
iii Random Search	5
iv Machine-learning Portfolios . .	5
iv.1 Erstes Modul „Predict20“ und „Predict1Year“	6
iv.2 Zweites Modul „PredictShares20“ und „PredictShares1Year“	7
iv.3 Drittes Modul MVM_new, MVM_modified, MVM1Year und MVM1Year_modified . .	7
v Portfolioallokation	8
III Anwendung	9
i Predict20 und Predict1Year . .	9
ii PredictShares20 und Predict1Year	10
iii MVM_new und MVM_modified	11
iv Reihenfolge der Module bei der Ausführung	11
v Statistische Module	12
IV Ergebnisse und Diskussion	12
i Ergebnisse	12
ii Vorhersagegenauigkeit	12
iii Renditen der Portfolios	12
iv Indexverlauf DAX:	12
v Statistische Modelle	12
vi ML-basierte Modelle	12
vii Modifizierte Mean-Varianz-Modelle	13
viii Sieger- und Verlierermodell des Vergleichs	14
ix Aktuelle Periode – Januar bis April 2024	14
x Diskussion	14
V Fazit und Ausblick	15

3	Pipeline / Flussdiagramm für die Modelle mit maschinellem Lernen	8
4	Pipeline / Flussdiagramm für die Vorhersage der Modelle mit maschinellem Lernen	9
5	Pipeline / Flussdiagramm für das MVM und MVM modified bei den Modellen mit maschinellem Lernen	10
6	Pipeline / Flussdiagramm für den Ablauf der Portfolioallokation	11
7	Die Standardabweichung der einzelnen Modelle im Vergleich . .	13
8	Renditen der Portfolios aus dem Jahre 2019	13
9	Renditen der Portfolios aus dem Jahre 2019	14

ABBILDUNGSVERZEICHNIS

1	Konzept und Aufbau dieser Arbeit	4
2	Pipeline / Flussdiagramm für die statistischen Portfolios . . .	6

I. EINFÜHRUNG

Im ersten Teil dieser Arbeit wurde das Mean-Variance-Modell von Harry Markowitz sowie grundlegende Konzepte des maschinellen Lernens vorgestellt. Der Fokus lag dabei auf der theoretischen Fundierung und Methodik der Portfoliooptimierung. Mithilfe von statistischen und maschinellen Methoden sollen Vorhersagen zu Aktienkursen getroffen und verschiedene Modelle zur Portfolioverwaltung entwickelt und miteinander verglichen werden.

Der zweite Teil der Arbeit knüpft nahtlos an diese theoretischen Grundlagen an und fokussiert sich auf die praktische Umsetzung der vorgestellten Konzepte. Ziel ist es, die im ersten Teil theoretisch erarbeiteten Modelle zu implementieren und deren Leistungsfähigkeit anhand realer Daten zu evaluieren. Hierbei stehen insbesondere die Genauigkeit der Kursvorhersagen und die Effizienz der Portfoliooptimierungsmodelle im Mittelpunkt.

Zu Beginn dieses zweiten Teils wird die Methodik konkretisiert und die Implementierung der maschinellen Lernalgorithmen sowie die Optimierungsverfahren beschrieben. Daraufhin erfolgt eine detaillierte Analyse der Ergebnisse, wobei die Performance der Modelle sowohl auf historische Daten als auch auf die Prognosen für das aktuelle Jahr 2024 überprüft wird. Abschließend werden die Vorhersagegenauigkeit und die einzelnen Renditen der Modelle bzw. Portfolios diskutiert.

Dieser praxisorientierte Ansatz soll aufzeigen, ob moderne Algorithmen des maschinellen Lernens eine Verbesserung gegenüber traditionellen statistischen Methoden in der Finanzmarktprognose und Portfoliooptimierung bieten können. Die gewonnenen Erkenntnisse dienen nicht nur der Validierung der theoretischen Modelle, sondern liefern auch wertvolle Hinweise für die zukünftige Anwendung und Weiterentwicklung in der Finanzpraxis.

Durch die Verknüpfung von Theorie und Praxis soll ein umfassendes Verständnis der Potentiale und Herausforderungen moderner Vorhersagemodelle und Portfoliooptimierungsmethoden vermittelt werden.

II. ENTWURF

Insgesamt wurden acht Modelle sowie der Dax betrachtet. Neu hinzugekommen sind zwei Modelle, die im ersten Teil der Untersuchung noch nicht vorhanden waren. Diese beiden neuen Modelle umfassen ein passiv verwaltetes und ein aktiv verwaltetes statistisches Modell, das auf einem modifizierten Mean-Variance-Modell basiert. Das Hinzufügen dieser statistischen Modelle war notwendig, um einen korrekten Vergleich mit den Modellen des maschinellen Lernens zu ermöglichen. Die vier statistischen Modelle wurden zudem in zwei Varianten ausgeführt: Einmal mit Daten, die die letzten vier Jahre umfassen, und einmal nur mit Daten aus dem letzten Jahr. Diese Differenzierung war wichtig, um die Auswirkungen unterschiedlicher Datenzeiträume auf die Modelle zu untersuchen. Für die Umsetzung der statistischen Modelle wurden vier Pythonskripte entwickelt:

1. Ein aktiv verwaltetes statistisches Modell mit Mean-Varianz-Portfoliooptimierung
2. Ein passiv verwaltetes statistisches Modell mit Mean-Varianz-Portfoliooptimierung
3. Ein aktiv verwaltetes statistisches Modell mit modifizierter Mean-Varianz-Portfoliooptimierung
4. Ein passiv verwaltetes statistisches Modell mit modifizierter Mean-Varianz-Portfoliooptimierung

Es ist generell anzumerken, dass aktiv verwaltete Modelle monatlich neu gewichtet werden und eine Laufzeit von einem Jahr haben. Im Gegensatz dazu werden passive Modelle nur einmal zu Beginn der Jahresperiode gewichtet und behalten diese Gewichtung für das gesamte Jahr bei. Zusätzlich wurden vier Modelle mit maschinellem Lernen (ML) implementiert:

1. Ein aktiv verwaltetes ML-Modell mit Mean-Varianz-Portfoliooptimierung
2. Ein passiv verwaltetes ML-Modell mit Mean-Varianz-Portfoliooptimierung
3. Ein aktiv verwaltetes ML-Modell mit modifizierter Mean-Varianz-Portfoliooptimierung

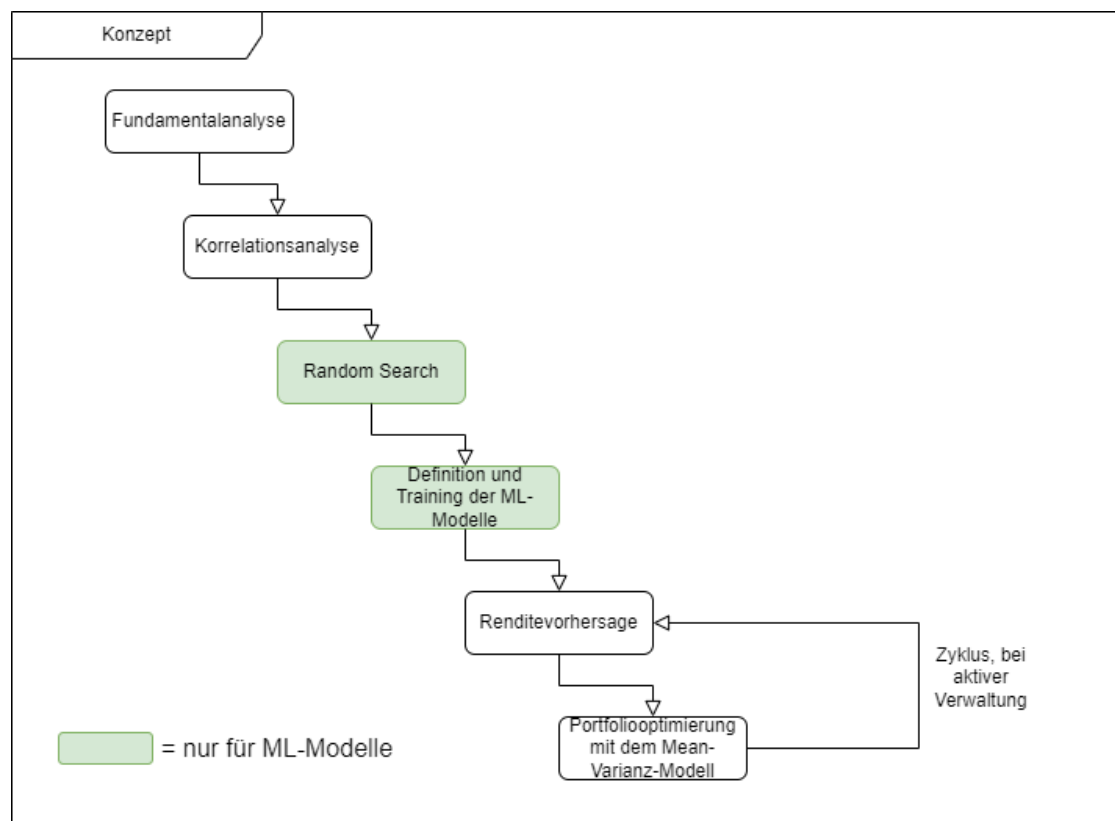


Abbildung 1: Konzept und Aufbau dieser Arbeit

4. Ein passiv verwaltetes ML-Modell mit modifizierter Mean-Varianz-Portfoliooptimierung

Wenn kein modifiziertes Mean-Variance-Modell verwendet wurde, kam selbstverständlich das normale Mean-Variance-Modell zum Einsatz. In der Abbildung 1 wurde der grundlegende Aufbau verbildlicht. Im Folgenden wird der Aufbau im Detail beschrieben.

i. Fundamentalanalyse

Für die Auswahl der Aktien wurde eine Fundamentalanalyse durchgeführt. Zunächst wurden pro Branche die drei Aktien mit dem niedrigsten Kurs-Gewinn-Verhältnis (KGV) ausgewählt. Im nächsten Schritt wurde die Korrelation der drei ausgewählten Aktien innerhalb jeder Branche geprüft, um eine Aktie zu eliminieren. Dabei wurden die zwei Aktien pro Branche ausgewählt, die die schwächste Korrelation aufwiesen. Dadurch blieben insgesamt 12 Aktien übrig, also zwei Aktien pro Branche.

ii. Statistische Portfolios

Zunächst wird bei den aktiv verwalteten Portfolios die .json-Datei mit den Profiten der letzten Periode eingelesen. Beim modifizierten Modell werden zusätzlich die Dividenden eingelesen und auf die Rendite addiert. Anschließend werden die Finanzdaten je nach Modell entweder für das letzte Jahr oder für die letzten vier Jahre heruntergeladen. Dabei werden der Eröffnungswert, der Höchstwert, der Tiefstwert, der Schlusskurs und das Volumen erfasst. Diese Daten werden auf fehlende Werte geprüft und gegebenenfalls durch den Mittelwert der vorherigen und der folgenden Zeile ersetzt. Danach wird der letzte Tag vor der vorherzugesagten Periode ermittelt, und die Schlusswerte dieses Tages werden extrahiert, um den Portfoliowert zu Beginn des Zyklus zu bewerten. Im nächsten Schritt werden die Finanzdaten der aktuellen Periode von Yahoo Finance heruntergeladen, und der Schlusswert des letzten Tages dieser Periode wird extrahiert, um den Wert des Portfolios am Ende des Zyklus be-

stimmen zu können. Die Daten aus den letzten vier Jahren beziehungsweise dem letzten Jahr werden in Blöcke von 20 bzw. 252 Dateneinträgen aufgeteilt, da 20 Werktage etwa einem Kalendermonat und 252 Werktage einem Kalenderjahr entsprechen. Denn es soll immer eine Vorhersage für einen Monat, bei den aktiv verwalteten Portfolios, und für ein Jahr, bei den passiv verwalteten Portfolios, erstellt werden. Pro Block wird die Rendite ermittelt, und aus diesen Renditen wird die durchschnittliche Rendite über alle Blöcke gebildet. Beim modifizierten Modell wird die Rendite der risikofreien Anlage hinzugefügt. Daraufhin wird die Standardabweichung über den gleichen Zeitraum errechnet, wobei beim modifizierten Modell die Standardabweichung von 0 der risikofreien Anlage hinzugefügt wird. Anschließend werden die Finanzdaten des letzten Jahres geladen, und es wird eine Datei mit den Schlusskursen aller betrachteten Aktien erstellt. Diese Daten sind für die Berechnung der Kovarianzmatrix wichtig, die während der Portfolioallokation zum Einsatz kommt. Nun werden die Mean-Varianz-Optimierung und die Portfolioallokation durchgeführt. Die Eingabeparameter für diese Funktionen umfassen die erwartete Rendite, die Standardabweichung und die Schlusskurse des letzten Jahres aller Aktien, um die Kovarianzmatrix zu erstellen. Danach wird die vorherzusagende Periode simuliert, wobei bestimmt wird, wie viele Anteile einer Aktie beziehungsweise der risikofreien Anlage gekauft werden müssen. Zudem wird der Wert des Portfolios bestimmt, um sicherzustellen, dass nur die 1000 € plus die Profite aus den letzten Perioden (nur bei den aktiv verwalteten Portfolios) verwendet werden. Zum Schluss werden die Aktienanteile mit dem Schlusskurs des vorherzusagenden Monats addiert, und der Wert des Portfolios wird ermittelt, um den Gewinn oder Verlust des Portfolios zu berechnen. Falls das Portfolio aktiv verwaltet wird, wird das Ergebnis, also der Profit, in eine .json-Datei gespeichert. Für das besser Verständnis wurde in Abbildung 2 der Entwurf des Codes in Form einer Pipeline bildlich dargestellt. Rot gefärbte Formen stellen dabei Code dar, der nur in den modifizierten Portfolios ausgeführt wird. Grün wird nur von Portfolios ausgeführt die aktiv verwaltet werden.

iii. Random Search

Um gute Ergebnisse zu erzielen, müssen bei machine-learning Modellen verschiedene Hyperparameter wie die Anzahl der Schichten, die Anzahl der Neuronen pro Schicht und die Lernrate eingestellt werden. Dies kann entweder durch manuelles Ausprobieren oder durch systematische Methoden zur Bewertung der verschiedenen Hyperparametereinstellungen erreicht werden. Hier kommt die Random-Search-Methode ins Spiel. Die Random-Search-Methode ist besonders geeignet, wenn das Modell nicht sehr komplex ist. Sie bewertet verschiedene zufällige Kombinationen von Parametern aus einem vorgegebenen Suchraum (vgl. Nair 2022). Im Vergleich zur Grid-Search-Methode, bei der alle Kombinationen systematisch ausprobiert werden, bietet Random Search mehrere Vorteile. Erstens ist sie effizienter und schneller (vgl. Nair 2022). Zweitens spürt sie gute Kombinationen schnell auf, ohne den gesamten Suchraum durchsuchen zu müssen (vgl. Nair 2022). Zudem vermeidet sie das häufige Testen vieler unnötiger Kombinationen, wie es bei der Grid-Search-Methode der Fall ist (vgl. Nair 2022). Allerdings hat die Random-Search-Methode auch Nachteile. Es gibt keine Garantie dafür, dass die beste Kombination gefunden wird (vgl. Nair 2022). Trotz dieses Nachteils bietet die Random-Search-Methode für dieses Forschungsprojekt die meisten Vorteile. Das Modell ist nicht besonders komplex, es steht keine große Rechenkapazität zur Verfügung, und es kann auch nicht viel Zeit aufgewendet werden. Das Random-Search-Modul ist ähnlich aufgebaut wie die Module „Predict20“ und „Predict1Year“. Der einzige Unterschied besteht in der Definition des Modells, bei der die Random-Search-Methode implementiert wurde. Vor der Definition wurden Parameterräume für die einzelnen Hyperparameter vorgegeben. Nach der Definition wird das Modell trainiert und bewertet, und schließlich wird das Modell mit der besten Bewertung ausgegeben.

iv. Machine-learning Portfolios

Die machine-learning Portfolios bestehen aus drei Modulen. Im ersten Modul werden die Modelle definiert und trainiert. Im zweiten Modul

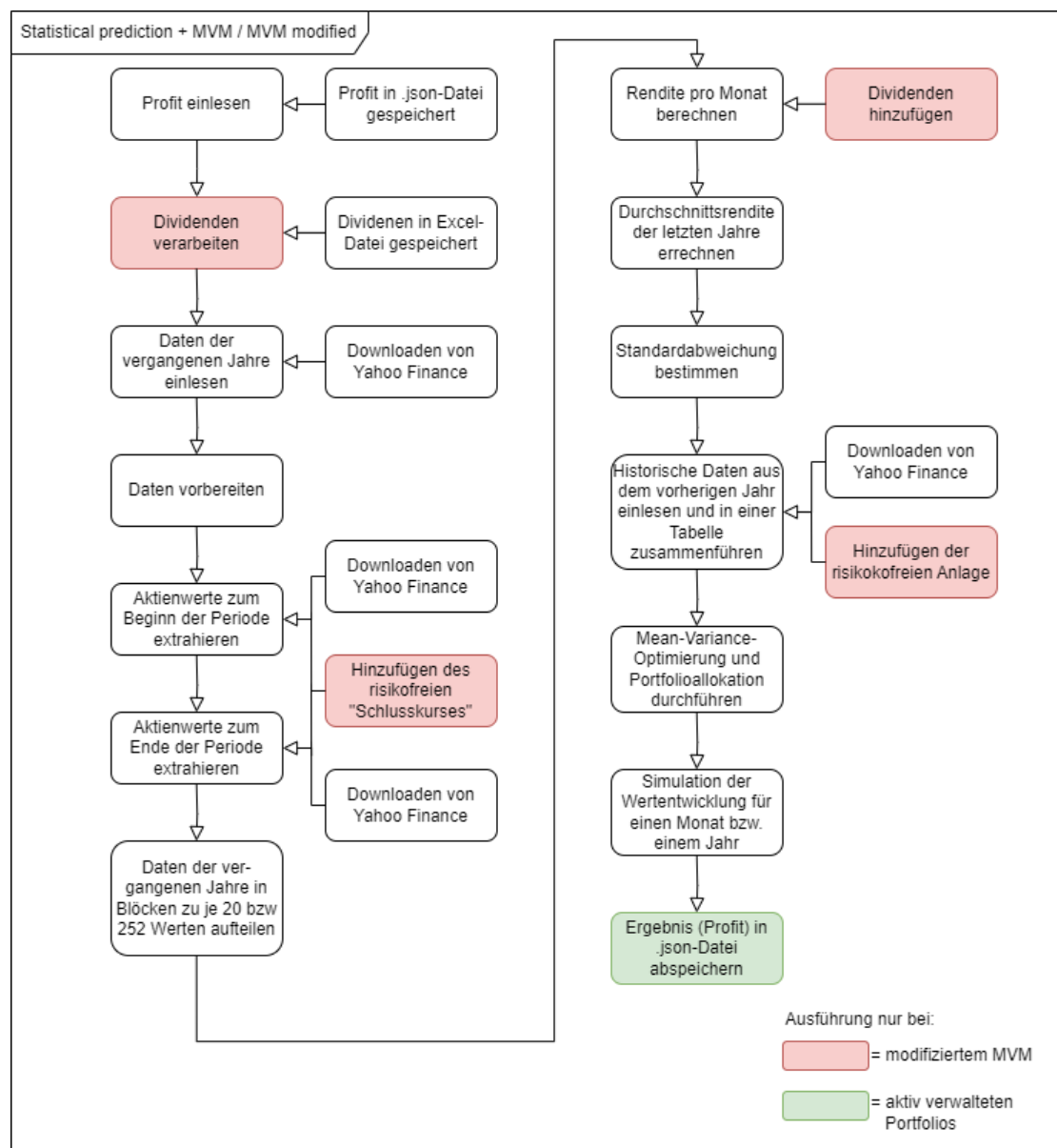


Abbildung 2: Pipeline / Flussdiagramm für die statistischen Portfolios

werden die Vorhersagen für die Aktienwerte ermittelt und im dritten Modul wird die Portfoliooptimierung durchgeführt und die vorherzusagende Periode simuliert.

iv.1 Erstes Modul „Predict20“ und „Predict1Year“

Zur besseren Veranschaulichung wurde die Module in der Abbildung 3 verbildlicht. Zunächst werden Trainingsdaten für jede Aktie aus dem Zeitraum von 2000 bis 2018 bzw. 2023 eingelesen, je nachdem, ob das Jahr 2019 oder 2024

vorhergesagt werden soll. Fehlende Werte werden, wie bei den statistischen Modellen, durch den Mittelwert der vorherigen und der folgenden Zeile ersetzt. Danach werden die Daten normalisiert. Die letzte Periode vor dem vorherzusagenden Zeitraum wird eingelesen und ebenfalls auf fehlende Werte überprüft und normalisiert. Anschließend werden Datensequenzen von 60 Datenpunkten erstellt. Für jeden Eingabewert „X“ wird der entsprechende Soll-ergebniswert, der 20 Tage (ein Monat für aktiv verwaltete Modelle) bzw. 252 Tage (ein Jahr für

passiv verwaltete Modelle) in der Zukunft liegt, angehängt, sodass das Modell lernt, den Wert 20 bzw. 252 Tage in der Zukunft vorherzusagen. Die Trainingsdaten werden in Trainings-, Test- und Validationsdaten aufgeteilt. Anhand der Validationsdaten wird die Leistung des Modells überprüft, um Overfitting oder Underfitting zu erkennen. Die Testdaten dienen der endgültigen Bewertung des Modells und überprüfen dessen Generalisierungsfähigkeit. Die Daten werden umgeformt und in Tensoren umgewandelt, um diese besser in das Modell einpflegen zu können. Danach werden das Modell und die Hyperparameter definiert, das Modell evaluiert und in einer externen Datei gespeichert. Schließlich wird der Testdatensatz überprüft, um die Vorhersagefähigkeit für eine unbekannte Periode zu bewerten. Eine weitere unbekannte Periode wird als Eingabedaten dem Modell übergeben, und die Vorhersage wird mit den tatsächlichen Werten verglichen. Um diesen Vergleich in monetären Werten durchführen zu können, müssen die normalisierten Werte wieder in monetäre Daten umgewandelt werden.

iv.2 Zweites Modul „PredictShares20“ und „PredictShares1Year

Wie aus Abbildung 4 entnommen werden kann, werden als erstes die Finanzdaten für die letzte Periode vor dem vorherzusagenden Zeitraum von Yahoo Finance heruntergeladen, wobei der Eröffnungswert, der Höchstwert, der Tiefstwert, der Schlusskurs und das Volumen erfasst werden. Die Daten werden auf fehlende Werte geprüft und gegebenenfalls durch den Mittelwert der vorherigen und der folgenden Zeile ersetzt. Danach werden die Daten normalisiert und vorbereitet, indem Dimensionen hinzugefügt und die Daten in Tensoren umgewandelt werden, um sie in das Modell einpflegen zu können. Anschließend werden die machine-learning Modelle importiert und die Vorhersage für den Aktienwert von dem jeweiligen Modell ermittelt und wieder zu monetären Werten umgewandelt. Die tatsächlichen Werte dieser Periode werden ebenfalls von Yahoo Finance heruntergeladen, und die Vorhersage jeder Aktie wird mit dem tatsächlichen Wert verglichen. Der quadratische Fehler zwischen Vorhersage und tatsächlichem Wert wird ermittelt und

für alle Aktien aufsummiert. Anschließend werden noch die Vorhersagen in eine .json-Datei geschrieben.

iv.3 Drittes Modul MVM_new, MVM_modified, MVM1Year und MVM1Year_modified

Auch hier wurden zur besseren Veranschaulichung die Module in der Abbildung 5 verbildlicht. Wieder stellen rot gefärbte Formen Code dar, der nur in den modifizierten Portfolios ausgeführt wird. Grün wird nur von Portfolios ausgeführt, die aktiv verwaltet werden. Zunächst werden die Daten der letzten Periode vor der vorherzusagenden Periode heruntergeladen. Der letzte Tag vor der vorherzusagenden Periode wird ermittelt, und die Schlusswerte dieses Tages werden extrahiert, um den Portfoliowert zu Beginn des Zyklus zu bewerten. Die Vorhersagen werden aus der .json-Datei eingelesen, und beim modifizierten Mean-Varianz-Modell wird die Dividende auf die Vorhersage addiert. Aus der Vorhersage wird die erwartete Rendite in Prozent berechnet, und bei den modifizierten Portfolios wird noch die Rendite der risikofreien Anlage hinzugefügt. Anschließend wird bei den aktiv verwalteten Portfolios der Profit der vergangenen Periode aus der .json-Datei eingelesen. Danach werden die Schlusskurse des letzten Tages der vorherzusagenden Periode aus Yahoo Finance heruntergeladen, um das Portfolio am Ende des Monats bestimmen zu können. Die Finanzdaten des letzten Jahres werden geladen und eine Datei mit den Schlusskursen aller betrachteten Aktien erstellt. Diese Daten sind für die Kovarianzmatrix wichtig, die während der Portfolioallokation zum Einsatz kommt. Bei dem modifizierten Mean-Variance-Modell wird die risikofreie Anlage hinzugefügt.

Danach werden die Mean-Variance-Optimierung und die Portfolioallokation durchgeführt. Diese Funktionen verwenden die Vorhersagen der machine-learning Modelle und die Schlusskurse des letzten Jahres aller Aktien als Eingabeparameter für die Kovarianzmatrix. Anschließend wird die vorherzusagende Periode simuliert, wobei bestimmt wird, wie viele Anteile einer Aktie bzw. der risikofreien Anlage gekauft werden müssen. Der Wert des Portfolios wird ermittelt, um sicherzustellen,

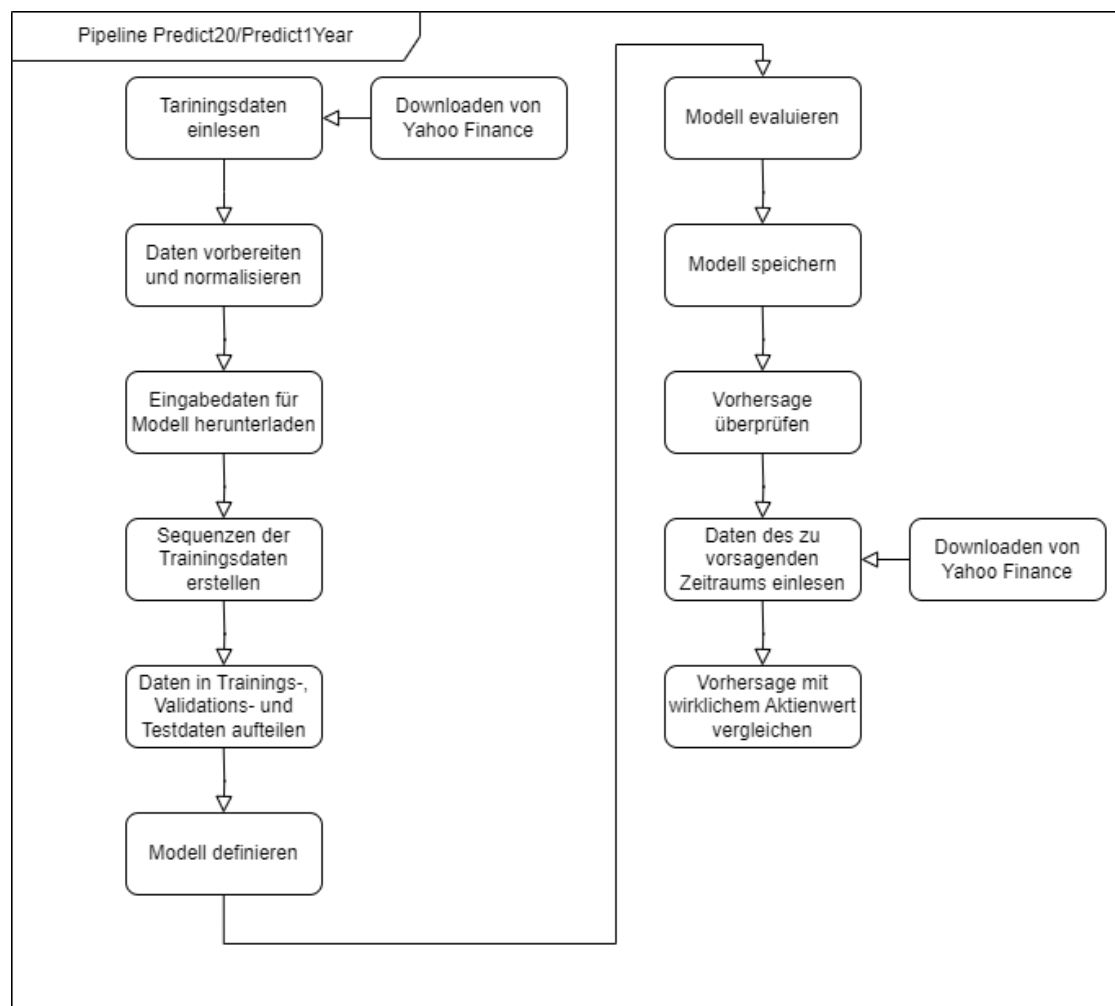


Abbildung 3: Pipeline / Flussdiagramm für die Modelle mit maschinellem Lernen

dass nur die 1000 € plus Profit (bei den aktiv veralteten Portfolios) aus den letzten Perioden verwendet wurden. Zum Schluss werden die Aktienanteile mit dem Schlusskurs des vorherzusagenden Monats addiert und der Wert des Portfolios wird ermittelt. Dadurch kann, wie bei den statistischen Modellen, der Gewinn oder Verlust des Portfolios ermittelt werden. Falls das Portfolio aktiv verwaltet wird, wird der Überschuss, also der Profit, in eine .json-Datei gespeichert. Ein negativer Profit würde einen Verlust darstellen.

v. Portfolioallokation

Das Flussdiagramm in Abbildung 6 illustriert den Prozess der Portfolioallokation im Rahmen der Mean-Variance-Optimierung bei allen Modellen. Nur bei den statistischen Modellen

wird zusätzlich die Standardabweichung in die Optimierung miteinbezogen. Der Prozess umfasst zwei Hauptfunktionen: `allocate_portfolio` und `mean_variance_optimization`, die zusammenarbeiten, um die optimale Portfolio-Gewichtung zu bestimmen. Zuerst werden in der Funktion `allocate_portfolio` die Vorhersagen der zukünftigen Aktienrenditen gemittelt, um eine einheitliche Erwartung zu erhalten. Anschließend wird die Kovarianzmatrix der historischen Daten berechnet. Diese Matrix stellt die Varianz und die Korrelationen zwischen den verschiedenen Aktien im Portfolio dar. Mit den erwarteten Renditen und der Kovarianzmatrix werden dann die optimalen Gewichtungen der Aktien, mithilfe der Funktion `mean_variance_optimization`, bestimmt. Dies erfolgt durch die Minimierung des Risikos und

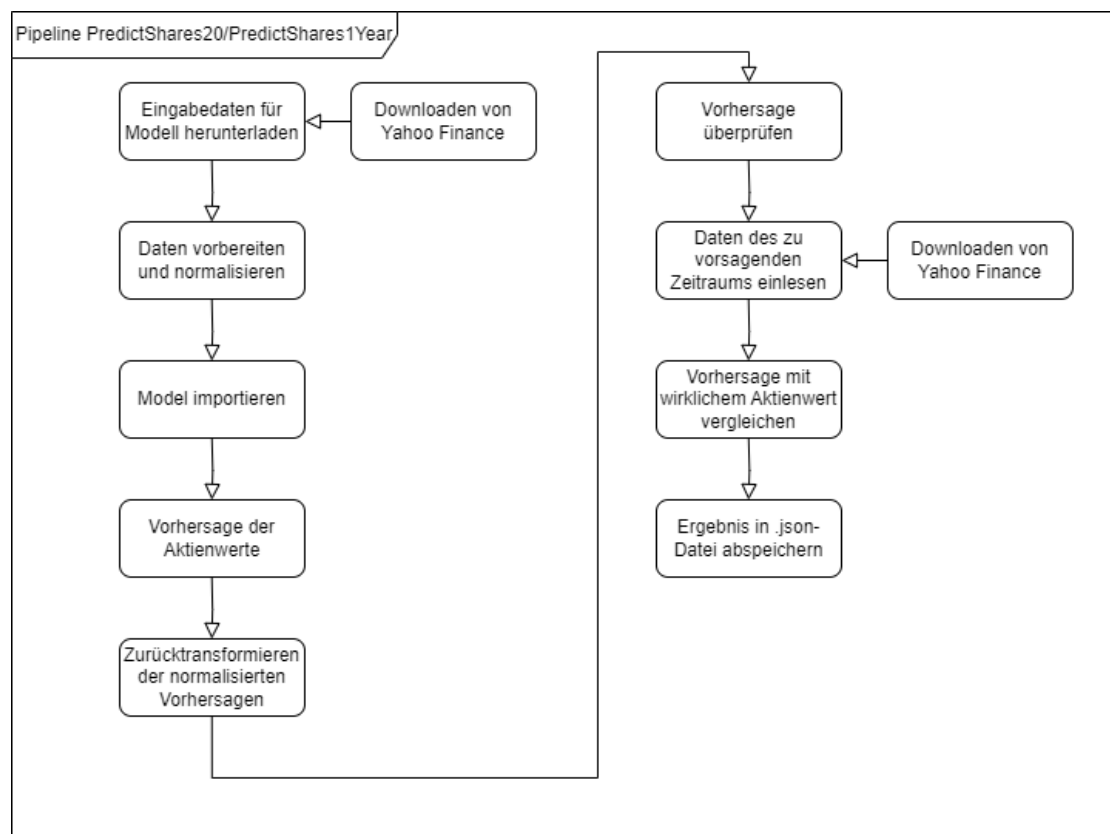


Abbildung 4: Pipeline / Flussdiagramm für die Vorhersage der Modelle mit maschinellem Lernen

Maximierung der Rendite gemäß der Mean-Variance-Theorie von Markowitz. Schließlich wird die berechnete Portfoliogewichtung zurückgegeben, die das Verhältnis der Investitionen in die verschiedenen Aktien darstellt.

Die Funktion `mean_variance_optimization` beginnt damit, ein Array zu erstellen, das die Gewichtungen der verschiedenen Aktien im Portfolio enthält. Die Gewichtungen werden auf Werte zwischen 0 und 1 beschränkt, um sicherzustellen, dass keine negativen Investitionen oder Leerverkäufe erfolgen. Das Hauptziel dieser Funktion ist es, die negative Sharpe-Ratio zu minimieren. Die Sharpe-Ratio misst die Rendite des Portfolios im Verhältnis zum Risiko. Durch Minimierung der negativen Sharpe-Ratio wird das Portfolio so optimiert, dass es die höchste risikoadjustierte Rendite bietet. Die Eingabedaten für die Funktion `allocate_portfolio` umfassen die Vorhersagen der zukünftigen Renditen der Aktien (`predictions`), die historischen Daten der Aktienkurse (`historical_data`) und das Anfangskapital, das

investiert werden soll (`initial_capital`). Für die Funktion `mean_variance_optimization` werden die erwarteten Renditen der Aktien (`expected_returns`), die aus den Vorhersagen abgeleitet werden, sowie die Kovarianzmatrix (`covariance_matrix`) benötigt. Diese beiden Funktionen arbeiten zusammen, um die optimale Portfoliogewichtung zu berechnen, indem sie die Vorhersagen und Kovarianzmatrix verwenden, um die Renditen zu maximieren und das Risiko zu minimieren.

III. ANWENDUNG

In diesem Kapitel wird erläutert, welche Parameter der Benutzer ändern muss, um selbst eine Portfoliooptimierung durchführen zu können.

i. Predict20 und Predict1Year

Bei den Modulen `Predict20` und `Predict1Year` muss der Benutzer beim Einlesen der CSV-

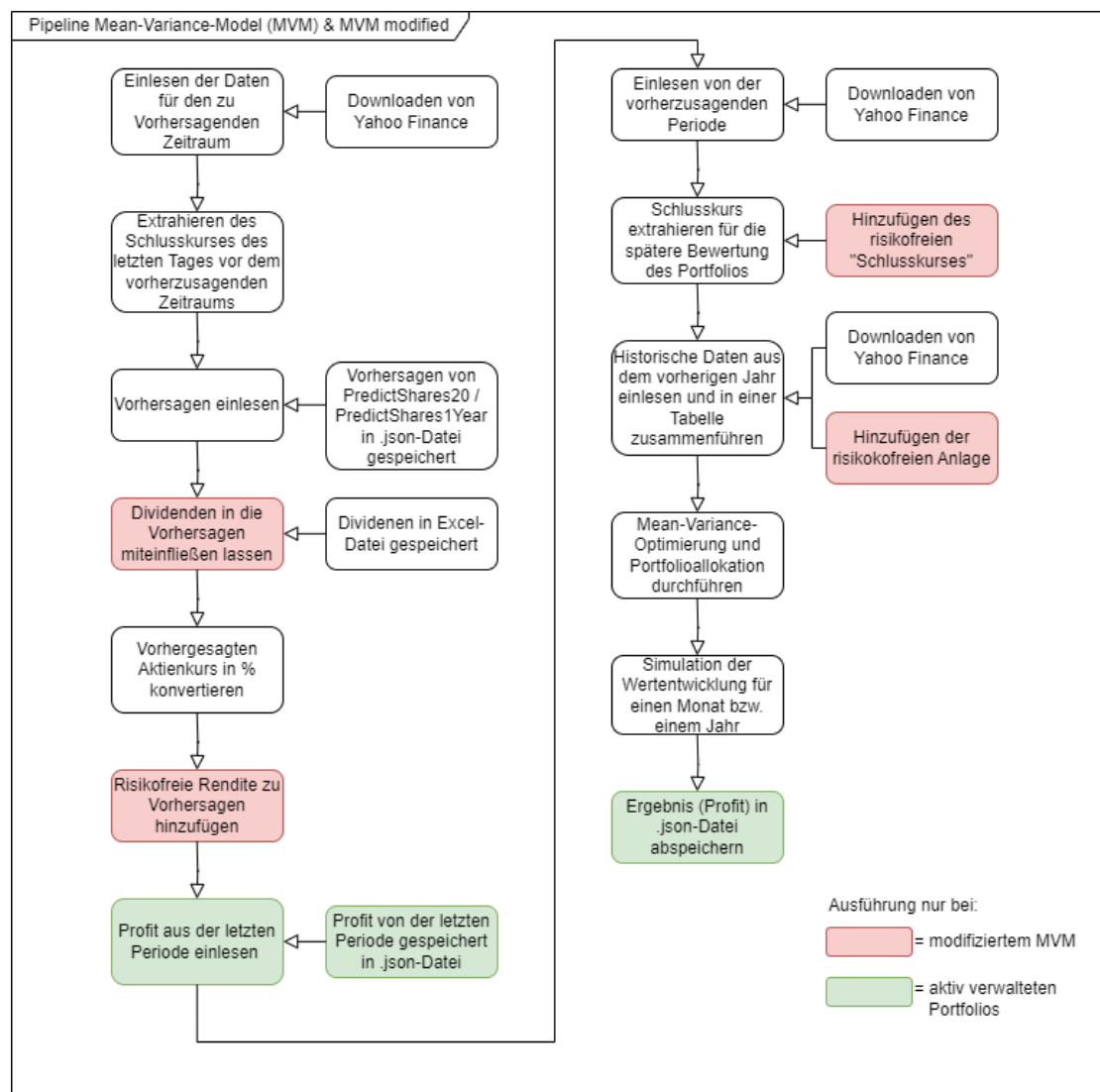


Abbildung 5: Pipeline / Flussdiagramm für das MVM und MVM modified bei den Modellen mit maschinellem Lernen

Datei das Aktienkürzel sowie den Zeitraum eingeben, mit welchen Daten das Modell trainiert werden soll. Wenn die Vorhersage des ML-Modells mit dem tatsächlichen Wert überprüft werden soll, muss der Zeitraum der vorherzusagenden Periode unter „read test CSV 2019“ eingetragen werden. Dieser Code wurde ursprünglich für die Aktienwertvorhersage für das Jahr 2019 geschrieben, kann jedoch für jede gewünschte Periode verwendet werden.

ii. PredictShares20 und Predict1Year

Für das Modul PredictShares20 sind keine Anpassungen im Code notwendig, es sei denn, es

sollen Vorhersagen für andere Aktiengemacht werden. In diesem Fall müssen die entsprechenden Aktienkürzel bei „stock_symbols“ unter „In [4]“ eingesetzt werden. Außerhalb des Codes sind jedoch ein paar Maßnahmen erforderlich: Der Ordner „Settings“ wird benötigt, und in diesem Ordner müssen die zwei .txt-Dateien auf 0 gesetzt werden, indem einfach eine 0 in die Datei geschrieben und diese gespeichert wird. Dies ermöglicht die Simulation des Jahres 2019. Sollten andere Jahre simuliert werden, müssen die Excel-Dateien entsprechend angepasst werden. Bei Predict1Year kann die letzte Periode vor der vorherzusagenden Periode unter der Überschrift „load time sequence“

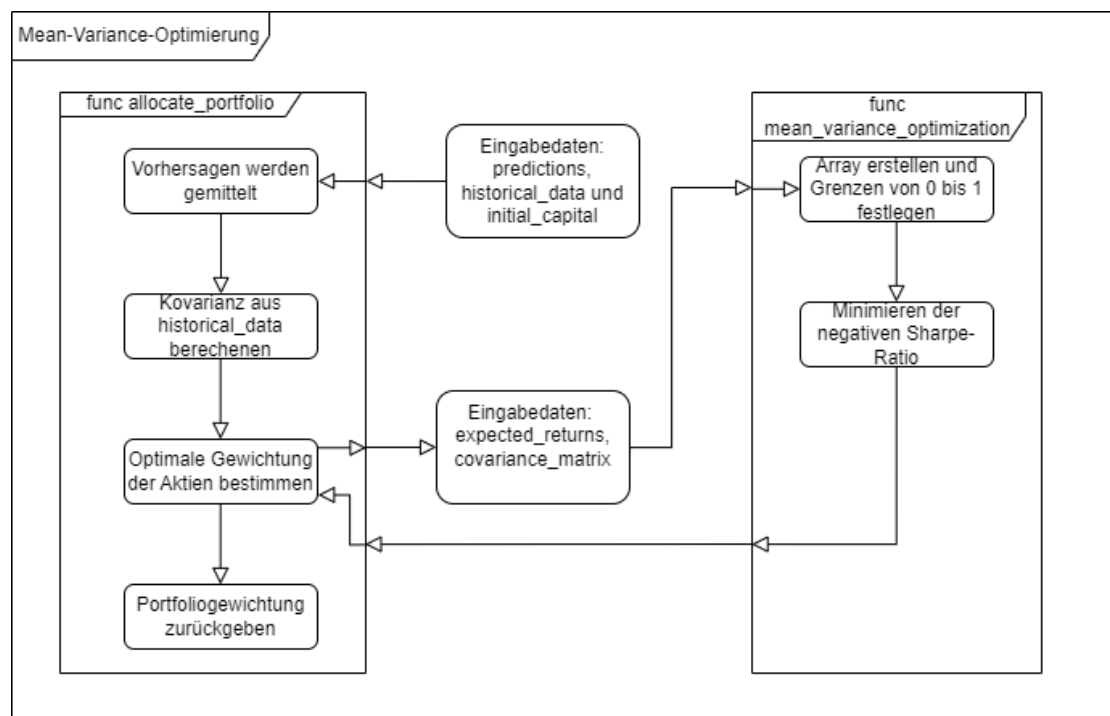


Abbildung 6: Pipeline / Flussdiagramm für den Ablauf der Portfolioallokation

in „In [3]“ angepasst werden. Das Modell sagt dann automatisch die nächste Periode, also das nächste Jahr (2019), voraus. Weiter unten kann zur Überprüfung der Vorhersage der tatsächliche Wert der Aktien geladen werden, indem die vorherzusagende Periode unter der Überschrift „read test CSV from 2019“ in „In [12]“ eingetragen wird.

iii. MVM_new und MVM_modified

In den Modulen MVM_new und MVM_modified können in „In [3]“ die Aktien angepasst werden. Bei MVM_new muss zu Beginn darauf geachtet werden, dass eine 0 in den .txt-Dateien eingetragen ist und dass keine profit.json-Datei vorhanden ist. Alle vorhandenen profit.json-Dateien sollten sicherheitshalber gelöscht werden. Bei MVM_modified muss in „In [8]“ die Excel-Datei mit den Dividenden des gewünschten Jahres ausgewählt werden. Zusätzlich kann in „In [6]“ die „risk_free_rate“, also die risikofreie Anlage, in Prozent eingetragen werden. In „In [13]“ muss das Startdatum für die Kovarianzmatrix eingetragen werden, wobei der Zeitraum immer ein Jahr umfassen

sollte. In den Modulen MVM1Year und MVM1Year_modified können die Aktien in „In [3]“ angepasst werden. In „In [4]“ muss der vorherzusagende Zeitraum eingetragen werden, und in „In [11]“ der Zeitraum für die Kovarianzmatrix, der das letzte Jahr vor der vorherzusagenden Periode darstellen sollte. Es ist wichtig, die Reihenfolge bei der Ausführung der Dateien zu beachten. Zuerst müssen die Module Predict20 bzw. Predict1Year ausgeführt werden, um die Modelle zu definieren. Danach folgen die Module PredictShares20 und Predict1Year, um die Vorhersagen zu ermitteln. Schließlich werden die Module MVM_new, MVM_modified, MVM1Year und MVM1Year_modified ausgeführt.

iv. Reihenfolge der Module bei der Ausführung

Für die Vorhersage eines Monats gehören **Predict20**, **PredictShares20** und **MVM_new** bzw. **MVM_modified** zusammen. **Predict20** und **Predict1Year** werden jedoch nur einmal am Anfang ausgeführt, um die ML-Modelle zu erstellen. Die Module **PredictShares20** und **MVM_new** bzw.

MVM_modified müssen dann zwölfmal wiederholt werden, um ein Jahr zu simulieren. Es sollte immer nur ein Modul, entweder **MVM_new** oder **MVM_modified**, bei einem Durchlauf (zwölf Wiederholungen) ausgeführt werden, da sonst die Einstellungen in der settings-Datei nicht mehr passen. Wenn beide Module ausgeführt werden sollen, sollte dies hintereinander erfolgen, wobei die settings nach dem ersten Durchlauf zurückgesetzt werden müssen. Um ein Jahr vorherzusagen, gehören **Predict1Year**, **PredictShares1Year** und **MVM1Year** bzw. **MVM1Year_modified** zusammen. Jedes Modul muss nur einmal ausgeführt werden, ohne dass weitere Anpassungen notwendig sind.

v. Statistische Module

Die statistischen Module werden genauso angepasst wie die ML-Module. Dabei ist zu beachten, dass sowohl die Vorhersage als auch die Portfoliooptimierung in einer einzigen Datei vorhanden sind. Anstatt separate Dateien für die Vorhersage (z. B. **PredictShares20**) und die Portfoliooptimierung (z. B. **MVM_new**) zu haben, existiert nur eine Datei für beide Aufgaben (z. B. **statistical20 MVM**).

IV. ERGEBNISSE UND DISKUSSION

i. Ergebnisse

In dieser Arbeit wurden die Renditen von acht verschiedenen Modellen sowie dem DAX über mehrere Zeiträume hinweg untersucht. Die Modelle umfassten sowohl traditionelle statistische Ansätze als auch Methoden des maschinellen Lernens, und wurden in aktiver sowie passiver Form betrieben. Zudem wurden zwei neue Modelle eingeführt, die auf einem modifizierten Mean-Variance-Modell basierten.

ii. Vorhersagegenauigkeit

In Abbildung 9 ist die Standardabweichung der Vorhersagen im Vergleich zu den tatsächlichen Aktienwerten dargestellt. Hierbei wurden jeweils fünf Perioden simuliert und deren Standardabweichungen berechnet, um anschließend den Durchschnitt dieser Werte zu ermitteln.

Die Grafik zeigt, dass die statistischen Vorhersagemodelle geringere Standardabweichungen aufweisen, was auf eine höhere Vorhersagegenauigkeit im Vergleich zu den maschinellen Lernmodellen hindeutet. Zudem haben Modelle, die den Aktienwert über eine einjährige Periode prognostizieren, deutlich höhere Standardabweichungen als jene, die nur eine einmonatige Periode vorhersagen.

iii. Renditen der Portfolios

Die Renditen der einzelnen Portfolios lassen sich aus den Abbildungen 8 und 9 entnehmen. Die wichtigsten Ergebnisse sind wie folgt:

Bei der Simulation der aktuellen Periode (Januar bis April 2024) konnten nur die aktiv verwalteten Modelle verwendet werden (siehe Abbildung 9), da die passiven Modelle einen Vorhersagezeitraum von einem Jahr benötigen.

iv. Indexverlauf DAX:

Der DAX erzielte eine Rendite von etwa 25 % und macht ihn somit äußerst konkurrenzfähig in der Periode 2019.

v. Statistische Modelle

Die aktiv und passiv verwalteten statistischen Modelle, mit Eingabedaten aus dem Jahr 2018, erzielten die höchsten Renditen von etwa 45 %. Das aktiv verwaltete Modell mit Daten bis 2018 erreichte ebenfalls eine hohe Rendite von etwa 45 %. Die Modelle, die mit Daten aus den Jahren 2014 bis 2018 trainiert wurden, schnitten deutlich schlechter ab und erzielten oft nahezu keine Rendite. Es könnte nun angenommen werden, dass weniger Eingabedaten schlechter abschneiden. Jedoch ist es bei der Periode Januar bis April 2024 genau anders herum. Dort haben die statistischen Modelle mit Eingabedaten aus vier Jahren besser abgeschnitten, als diese, die nur ein Jahr an Eingabedaten erhalten hatten.

vi. ML-basierte Modelle

Die ML-Modelle erzielten durchweg niedrigere Renditen im Vergleich zu den besten statistischen Modellen. Das aktiv und passiv verwal-

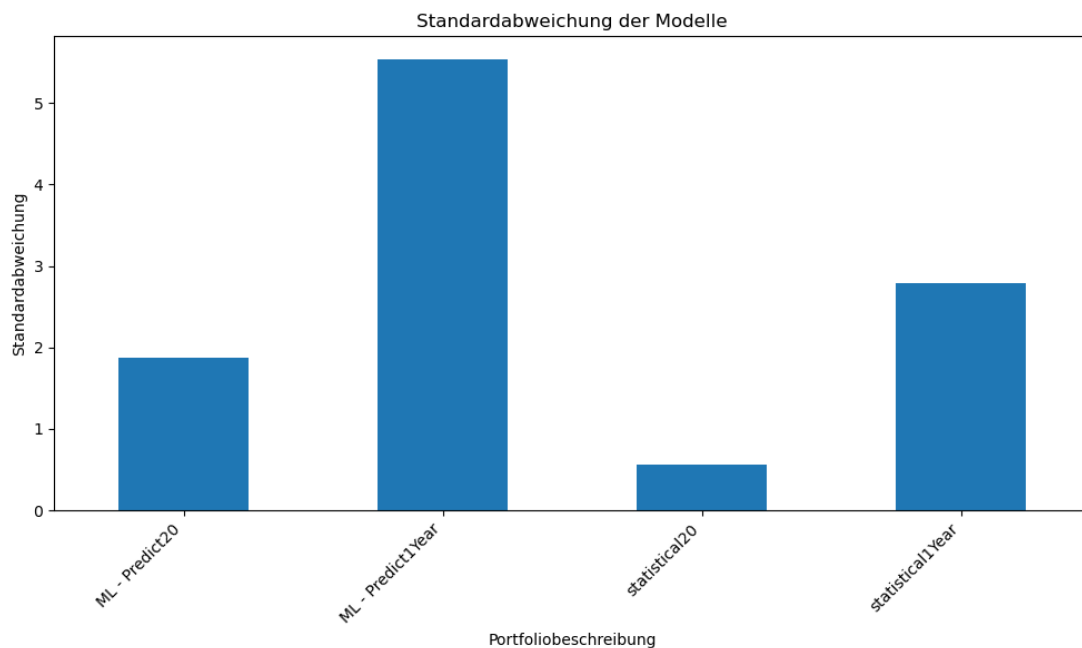


Abbildung 7: Die Standardabweichung der einzelnen Modelle im Vergleich

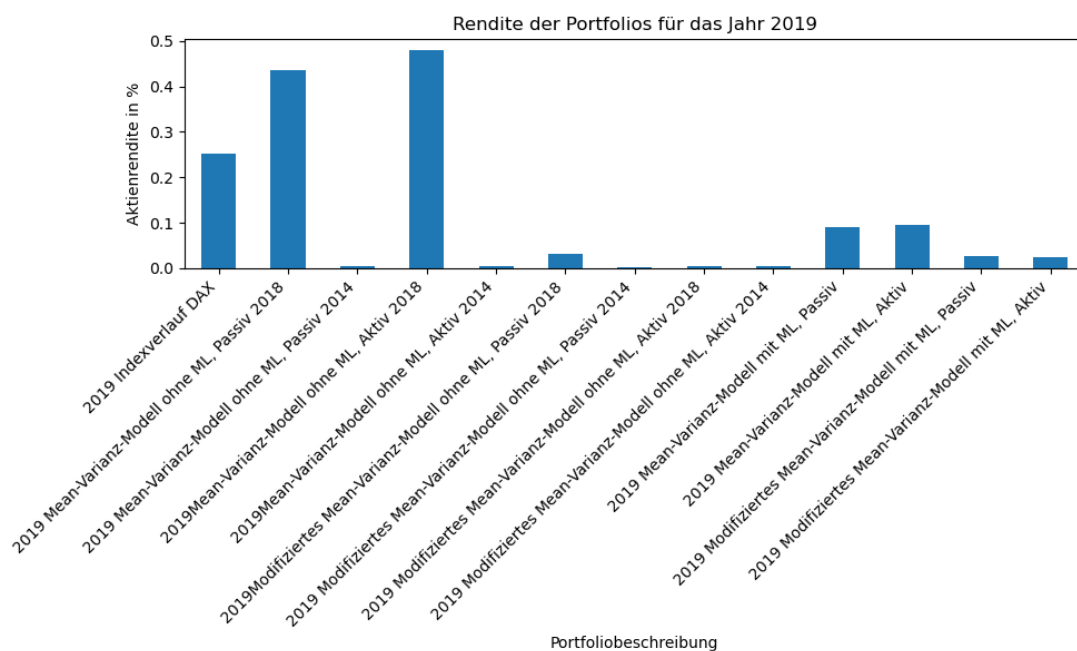


Abbildung 8: Renditen der Portfolios aus dem Jahre 2019

tete ML-Modell, mit normalem Mean-Varianz-Portfolio, erzielte die beste Rendite innerhalb der ML-Modelle von etwa 9 %. Auch fällt auf, dass es bei den ML-Modellen keinen großen Unterschied zwischen aktiv und passiv verwalteten Modellen bzw. Portfolios gibt.

vii. Modifizierte Mean-Varianz-Modelle

Die modifizierten Mean-Varianz-Portfolios haben durchweg die schlechtesten Ergebnisse erzielt. Mit ML waren diese im Schnitt etwas

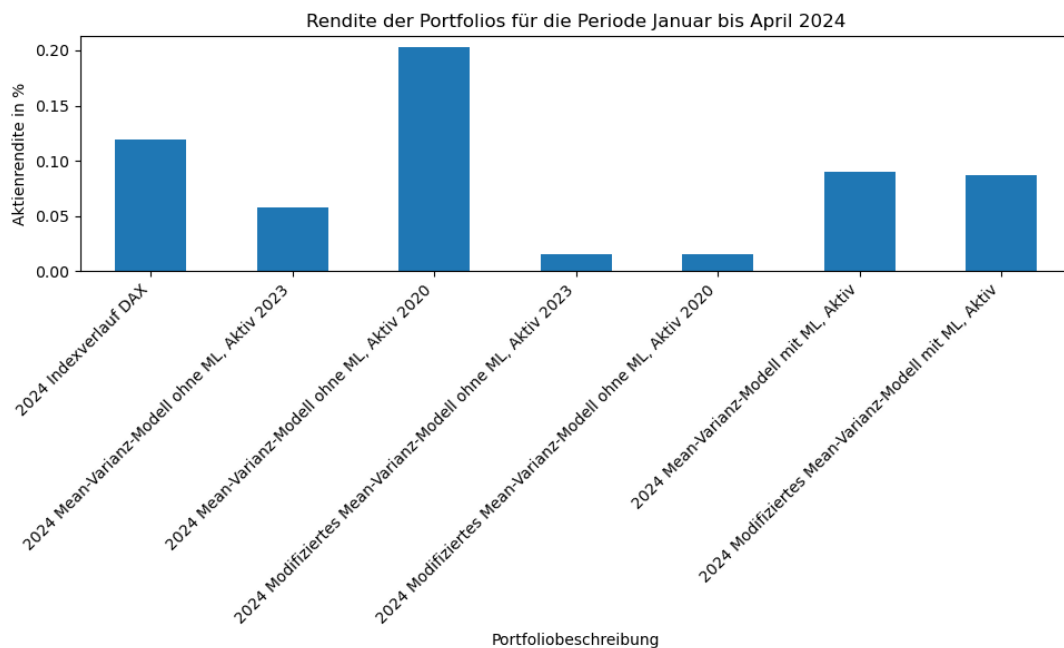


Abbildung 9: Renditen der Portfolios aus dem Jahre 2019

besser, aber auch noch viel schlechter, als die Modelle, welche das normale Mean-Varianz-Modell implementiert hatten.

viii. Sieger- und Verlierermodell des Vergleichs

Das erfolgreichste Modell war das aktiv verwaltete statistische Modell, ohne ML, mit Eingabedaten aus dem Jahr 2018, dass eine Rendite von 47 % erzielte.

Im Gegensatz dazu schnitten die modifizierten Portfoliomodelle nahezu durchgehend am schlechtesten ab. Dies deutet auf deren ineffektive Vorhersagefähigkeit hin.

ix. Aktuelle Periode – Januar bis April 2024

In der aktuellen Periode wurden nur aktiv verwaltete Modelle verwendet, da die passiven Modelle eine Periode von einem Jahr vorhersagen und hier nur ein Zeitraum von vier Monaten vorhanden war. Dort hat ebenfalls ein statistisches Modell die Nase vorne. Das aktiv verwaltete statistische Modell mit Eingabedaten von 2020 bis 2023 hat die meiste Rendite erzielt und schlägt damit auch den DAX. Trotzdem

ist anzumerken, dass in der aktuellen Periode das statistische Modell am erfolgreichsten war, das mit mehr Eingabedaten gefüttert wurde. Bei der Periode 2019 war dies umgekehrt. Die ML-Modelle haben in der aktuellen Periode auch wieder etwa 9 % erwirtschaftet. Damit ist die Leistung jedoch schlechter als der Dax mit knapp 12 %. Die schlechtesten Portfolios waren die modifizierten statistischen Mean-Varianz-Modelle mit gerade einmal 1,5 %.

x. Diskussion

Die Ergebnisse dieser Analyse zeigen deutlich, dass traditionelle statistische Modelle ohne den Einsatz von maschinellem Lernen in bestimmten Fällen bessere Renditen erzielen können als moderne ML-basierte Modelle. Dies bestätigen auch die Standardabweichungen der Vorhersagegenauigkeit aus Abbildung 9. Jedoch waren hier 2019, Eingabedaten von einer einjährigen Periode (2018) im Vorteil, im Gegensatz zu den Modellen, die Eingabedaten der Jahre 2014 bis 2018 erhalten hatten. In der aktuellen Periode war es jedoch genau umgekehrt der Fall. Nun haben die Modelle besser abgeschnitten, die Eingabedaten aus den Jahren 2020 bis 2023 erhalten hatten, im Gegensatz zu den Model-

len, die nur das Jahr 2023 als Eingabedaten erhalten haben. Dies zeigt die Unvorhersehbarkeit des Aktienmarktes, sodass nicht genau gesagt werden kann, welcher Eingabebereich bei den statistischen Modellen ein Vorteil darstellt. Die ML-Modelle mit normalen Mean-Varianz-Portfolio haben durchgehend in etwa 9 % Rendite eingefahren. Das ist zwar schlechter als der Dax und einige statistische Portfolios, jedoch wurde eine gute Rendite in beiden Perioden erwirtschaftet. Das zeigt, dass es in die richtige Richtung geht und diese Modelle etwas stabilere Renditen liefern, als die statistischen Modelle. Für die tendenziell niedrigeren Renditen könnte es verschiedene Gründe geben:

1. Datenqualität und -quantität: ML-Modelle sind oft sehr datenhungrig und benötigen große Mengen an qualitativ hochwertigen Daten. Es ist möglich, dass die verfügbaren Daten nicht ausreichen, um die Modelle effektiv zu trainieren.
2. Modellkomplexität und Overfitting: Die Gefahr des Overfitting ist bei ML-Modellen immer präsent, insbesondere wenn die Trainingsdaten nicht repräsentativ für zukünftige Daten sind. Dies könnte zu einer schlechten Generalisierungsfähigkeit geführt haben.

Außerdem macht es keinen großen Unterschied, ob die Portfolios aktiv oder passiv verwaltet wurden. Das widerspricht etwas der Abbildung 9. Es könnte daran liegen, dass eine Standardabweichung von fünf auf das Jahr gesehen nicht so einen großen Einfluss hat, im Gegensatz zu Vorhersagen für kürzere Perioden. Das zeigt, dass die ML-Modelle die Tendenz auch über lange Perioden ungefähr vorhersagen können. Die modifizierten Modelle haben durchschnittlich am schlechtesten abgeschnitten. Dies kann die folgenden Gründe gehabt haben:

1. Die Dividende macht auf den Monat gerechnet und auch auf das Jahr gerechnet kaum einen Unterschied in der Vorhersage. Wenn bedacht wird, dass die höchsten Dividenden bei etwa 9 bis 11 € pro Aktie lagen und die meisten noch deutlich darunter, macht das auf den Monat gerechnet keinen großen Unterschied bei den Renditevorhersagen aus. Deswegen wird dieser Faktor vernachlässigt.

2. Der zweite und entscheidendere Grund ist die risikolose Anlage, die inkludiert wurde. Hier konnte das Geld auf der Bank ohne Risiko angelegt werden. Durch die Standardabweichung von Null bei der risikolosen Anlage, wurde das Ergebnis der Portfolioallokation entscheidend beeinflusst. Dadurch wurde eine unverhältnismäßig große Menge des Startkapitals in die risikolose Anlage investiert, was zu einem geringen Risiko geführt hat, aber eben auch zu einer geringen Rendite. In zukünftigen Forschungen könnte versucht werden einen Höchstsatz für die risikolose Anlage zu implementieren, sodass nicht der Großteil des Geldes in die risikolose Anlage fließt.

V. FAZIT UND AUSBLICK

In dieser Arbeit wurden die Renditen und die Vorhersagegenauigkeit von acht verschiedenen Modellen sowie dem DAX über mehrere Zeiträume hinweg untersucht. Die Modelle umfassten sowohl traditionelle statistische Ansätze als auch Methoden des maschinellen Lernens, und wurden in aktiver sowie passiver Form betrieben. Zudem wurden zwei neue Modelle eingeführt, die auf einem modifizierten Mean-Variance-Modell basierten.

Die Arbeit zeigt deutlich, dass traditionelle statistische Modelle ohne den Einsatz von maschinellem Lernen in bestimmten Szenarien bessere Renditen erzielen können als moderne ML-basierte Modelle. Generell hatten die statistischen Modelle in diesem Anwendungsfall auch geringere Standardabweichungen, was für eine bessere Genauigkeit dieser spricht. Besonders hervorzuheben ist das aktiv verwaltete statistische Modell mit Eingabedaten aus dem Jahr 2018, welches mit einer Rendite von 47 % das erfolgreichste Modell war. Im Gegensatz dazu schnitten die modifizierten Portfoliomodelle durchgehend am schlechtesten ab, was deren ineffektive Portfolioallokation verdeutlicht. Die Analyse der Ergebnisse ergab jedoch, dass der Erfolg statistischer Modelle stark vom betrachteten Eingabezeitraum abhängt. Während in der Periode 2019 Modelle mit einjährigen Eingabedaten am besten abschnitten, erzielten in der aktuellen Periode (Januar bis April 2024) Modelle mit längeren Eingabezeiträumen die

besten Ergebnisse. Dies unterstreicht die Unvorhersehbarkeit des Aktienmarktes und macht es schwierig, eine eindeutige Empfehlung für den optimalen Eingabezeitraum zu geben.

ML-Modelle erzielten durchweg niedrigere Renditen als die besten statistischen Modelle, konnten jedoch stabile Erträge von etwa 9 % in beiden Perioden erwirtschaften. Dies deutet darauf hin, dass ML-Modelle, trotz ihrer geringeren Spitzenrenditen, eine gewisse Stabilität bieten könnten. Die geringeren Renditen der ML-Modelle könnten auf Herausforderungen wie unzureichende Datenqualität und -quantität sowie Modellkomplexität und Overfitting zurückzuführen sein. Außerdem konnte beobachtet werden, dass es keinen großen Unterschied macht, ob das Portfolio monatlich oder nur einmal im Jahr angepasst wird.

Sowohl bei den statistischen Modellen als auch bei den maschinellen Lernmodellen waren die Vorhersagen für den Zeitraum eines Jahres weniger präzise als für den Zeitraum eines Monats. Dennoch erzielten die Portfolios in beiden Fällen vergleichbar gute Ergebnisse. Dies könnte darauf zurückzuführen sein, dass die leicht erhöhte Standardabweichung über das Jahr hinweg keine signifikante Auswirkung hatte.

Die modifizierten Mean-Varianz-Modelle lieferten insgesamt die schlechtesten Ergebnisse, ganz gleich, ob die Vorhersagen mit statistischen Methoden oder mit ML-Modellen gemacht wurden. Dies ist vor allem auf zwei Faktoren zurückzuführen: die vernachlässigbare Auswirkung von Dividenden auf die monatliche und jährliche Vorhersage und die Überinvestition in risikolose Anlagen aufgrund deren Standardabweichung von Null.

Künftige Forschungen könnten versuchen, die modifizierten Portfoliomodelle zu verbessern, indem sie beispielsweise einen Höchstsatz für risikolose Anlagen implementieren. Außerdem könnten die ML-Modelle mit mehr Daten trainiert werden. Zudem könnten die Hyperparameter noch genauer, bspw. mit der Grid-Search-Methode, ermittelt werden. Für diese Methode waren in diesem Projektumfang weder Zeit- noch Rechenaufwand verhältnismäßig. Ein weiterer interessanter Ansatz wäre, Nachrichten über Aktienunternehmen, bspw. auf X (ehemals Twitter), zu analysieren und diese

Daten mit in das LSTM-Netzwerk einfließen zu lassen. Dabei könnten die Nachrichten in positive und negative Nachrichten eingeteilt werden, sodass der Aktienkurs vom LSTM-Netzwerk entsprechend nach oben oder nach unten korrigiert werden könnte.

Zusammenfassend zeigt diese Arbeit, dass traditionelle statistische Modelle in bestimmten Fällen überlegen sein können, während ML-Modelle tendenziell stabilere, wenn auch niedrigere Renditen liefern. Dazu muss erwähnt werden, dass die passive Investition in den DAX auch für die beiden Zeiträume die modifizierten Portfolios, die ML-Modelle und eine Großzahl der statistischen Modelle übertroffen hat. Die Ergebnisse betonen die Notwendigkeit weiterer Forschung, um die Vorhersagekraft und Robustheit beider Ansätze zu verbessern.

LITERATURVERZEICHNIS

Nair, Aashish (2022): Grid Search VS Random Search VS Bayesian Optimization, in: Medium, 07.05.2022, [online] <https://towardsdatascience.com/grid-search-vs-random-search-vs-bayesian-optimization-2e68f57c3c46>.

AUTORENANGABEN

Samuel Hessberger
Hochschule Fulda – Fachbereich Elektrotechnik und Informationstechnik
Leipziger Straße 123
36037 Fulda
Samuel-Lukas.Hessberger@hs-fulda.de