

Wann Multiple Testing und Data Snooping bias unkontrollierbar werden beim Trading mit ML am besonderen Beispiel von FP-Growth

nick.albrecht

January 2026

1 Introduction

[a4paper,11pt]article

amsmath,amssymb,amsthm booktabs geometry microtype hyperref margin=2.5cm

Kombination von Suchraum-Eskalationsmechanismen im Trading:

Wie die effektive Anzahl getesteter Varianten K explodiert und den Selektionsbias verstärkt

2 Ziel und Leitidee

In empirischem Trading-Research werden typischerweise viele Modellvarianten, Feature-Sets, Regeln, Schwellenwerte und Pipeline-Designs ausprobiert. Wird am Ende die beste Variante ausgewählt (z. B. nach Sharpe Ratio, annualisierter Rendite oder Utility), entsteht ein systematischer Selektionsbias (“Winner’s Curse”): selbst ohne echten Edge wird im Mittel eine scheinbar gute Performance gefunden.

Unter stark vereinfachenden Annahmen (kein echter Edge, stabile Verteilung, keine Korrelationen zwischen Varianten) lässt sich der erwartete Extremwert-Bias durch

$$\text{Bias} \approx \sqrt{\frac{\ln K}{T}} \quad (1)$$

beschreiben (Konstanten wie $\sqrt{2}$ sind für die Wachstumsdiskussion zweitrangig). Dabei ist:

- K : effektive Anzahl getesteter Varianten (Modelle/Regeln/Parameter-Setups etc.),
- T : Stichprobengröße (z. B. $T = 252$ bei einem Jahr täglicher Daten).

Der Schlüssel ist: *Der Bias wächst nicht “schnell” in K selbst, sondern in $\ln K$.* In der Praxis eskaliert der Bias dennoch sehr schnell, weil K als Funktion der Freiheitsgrade (Features, Regeln, Filter, Pipeline-Stufen) oft *kombinatorisch oder exponentiell* wächst. Dann wird $\ln K$ groß und die rechte Seite von (1) nimmt stark zu.

Dieses Dokument erklärt detailliert, wie K explodiert, wenn mehrere Eskalationsmechanismen kombiniert werden, und wie sich dies in Landau-Symbolen ausdrücken lässt.

3 Grundmodell: Selektion als Extremwertproblem

3.1 Performance-Schätzer und Selektion

Es werde eine Performancemetrik $\widehat{M}_k$ für Variante k betrachtet, z. B. Sharpe Ratio $\widehat{SR}_k$ oder eine standardisierte Teststatistik Z_k .

Ein analytisch einfaches Nullmodell lautet:

$$Z_k \sim \mathcal{N}(0, 1), \quad k = 1, \dots, K, \quad (2)$$

und es wird die beste Variante selektiert:

$$Z_{\max} = \max_{1 \leq k \leq K} Z_k. \quad (3)$$

Dann gilt (Extremwert-Näherung):

$$\mathbb{E}[Z_{\max}] \approx \sqrt{2 \ln K}. \quad (4)$$

3.2 Übersetzung in Sharpe und Rendite

Eine gängige grobe Näherung lautet unter der Null:

$$\widehat{SR}_k \approx \mathcal{N}\left(0, \frac{1}{T}\right), \quad \Rightarrow \quad \sqrt{T} \widehat{SR}_k \approx \mathcal{N}(0, 1). \quad (5)$$

Damit folgt aus (4):

$$\mathbb{E}[\widehat{SR}_{\max}] \approx \frac{\sqrt{2 \ln K}}{\sqrt{T}} = \Theta\left(\sqrt{\frac{\ln K}{T}}\right). \quad (6)$$

Mit (annualisiert) $SR = \mu/\sigma$ folgt für die Verzerrung der erwarteten Rendite:

$$\Delta\mu \approx \sigma \cdot \mathbb{E}[\widehat{SR}_{\max}] = \Theta\left(\sigma \sqrt{\frac{\ln K}{T}}\right). \quad (7)$$

Wichtig. In der gesamten weiteren Diskussion wird T als *nicht das Problem* betrachtet (konstant und hinreichend groß), und es wird ausschließlich analysiert, wie unvorsichtige Kombinationen von Freiheitsgraden K so stark wachsen lassen, dass sich die Wachstumsordnung in den Freiheitsgraden qualitativ verschlechtert.

4 Wie K eskaliert: Bausteine und Landau-Formen

In vielen praktischen Forschungsprozessen setzt sich die effektive Suchgröße K aus mehreren Komponenten zusammen. Typischerweise gilt entweder:

1. Produktform (Pipeline-Produkt):

$$K_{\text{all}} \approx \prod_{i=1}^m K_i,$$

2. Kombinatorische Form (Subset/Regel-Kombinationen):

$$K \sim 2^p \quad \text{oder} \quad K \sim (1 + L)^m,$$

3. Verschachtelte Form (outer $\times$ inner): z. B. Feature-Subsets außen und pro Subset Regelmining innen.

Im Folgenden werden einzelne Eskalationsmechanismen eingeführt, jeweils mit (i) konkreter K -Formel und (ii) Landau-Argumentation für $\ln K$.

4.1 Baustein A: Feature-Subset-Suche (Wrapper-Selection)

Beschreibung. Es existieren p potenzielle Features (Indikatoren, Lags, Transformationen). Wenn erlaubt ist, *beliebige Teilmengen* zu verwenden, dann ist die Anzahl möglicher Feature-Sets:

$$K_{\text{feat}} = 2^p. \quad (8)$$

Landau. Aus (8) folgt:

$$\ln K_{\text{feat}} = \ln(2^p) = p \ln 2 = \Theta(p). \quad (9)$$

Setzt man dies in (6) ein (bei konstantem T), erhält man:

$$\text{Bias} = \Theta\left(\sqrt{\frac{p}{T}}\right). \quad (10)$$

Interpretation. Hier kippt die Wachstumslogik: nicht mehr $\sqrt{\ln(\text{Trials})}$, sondern $\sqrt{p}$ in der Feature-Anzahl (weil K exponentiell in p ist). Schon moderate Erhöhungen von p vergrößern $\ln K$ linear.

4.2 Baustein B: Filter-/Regel-Kombinatorik mit Schwellen

Beschreibung. Es gebe m mögliche Filter/Conditions (z. B. “nur handeln wenn Volatilität $< x$ ”, “nur wenn Trendfilter true” etc.). Wenn beliebige Kombinationen erlaubt sind, ist die Anzahl möglicher Filterkombinationen:

$$K_{\text{filter}} = 2^m. \quad (11)$$

Wenn zusätzlich jeder Filter einen Schwellenwert aus einem Gitter mit L Optionen hat, ergibt sich (als grobe Kombinationszahl):

$$K_{\text{filter+thr}} = \sum_{j=0}^m \binom{m}{j} L^j = (1 + L)^m. \quad (12)$$

Landau. Aus (11) folgt $\ln K = \Theta(m)$, aus (12) folgt:

$$\ln K_{\text{filter+thr}} = \ln((1 + L)^m) = m \ln(1 + L) = \Theta(m \ln(1 + L)). \quad (13)$$

Damit:

$$\text{Bias} = \Theta\left(\sqrt{\frac{m \ln(1 + L)}{T}}\right). \quad (14)$$

Interpretation. Jeder zusätzliche Filter (oder eine Filterfamilie) erhöht m . Schon bei moderatem L ist die Suchraumvergrößerung massiv, da $(1 + L)^m$ exponentiell in m ist.

4.3 Baustein C: Mehrstufige Pipeline (Produkt der Stufen)

Beschreibung. In realen Pipelines werden mehrere Designentscheidungen kombiniert:

- Preprocessing-Variante: K_{pre}
- Target/Horizon: K_{tar}
- Modellfamilie: K_{mod}
- Hyperparameter-Trials: K_{hpo}
- Trading-Threshold/Position-Sizing: K_{trade}

Dann ist die Kombinationszahl typischerweise:

$$K_{\text{pipe}} \approx K_{\text{pre}} \cdot K_{\text{tar}} \cdot K_{\text{mod}} \cdot K_{\text{hpo}} \cdot K_{\text{trade}}. \quad (15)$$

Landau. Logarithmieren liefert:

$$\ln K_{\text{pipe}} \approx \ln K_{\text{pre}} + \ln K_{\text{tar}} + \ln K_{\text{mod}} + \ln K_{\text{hpo}} + \ln K_{\text{trade}}. \quad (16)$$

Somit addieren sich Suchraum-Beiträge in $\ln K$ direkt.

Interpretation. Selbst wenn jede Stufe für sich moderat ist, kann das Produkt groß werden. Kritisch wird es, wenn *eine* Stufe kombinatorisch ist (z. B. $K_{\text{trade}} \sim (1 + L)^m$ oder $K_{\text{feat}} \sim 2^p$), weil dann ein linearer Term in p oder m in $\ln K$ dominiert.

4.4 Baustein D: Objective-/Metrik- und Constraint-Hacking (Prozedurenraum)

Beschreibung. Wenn nicht nur Modelle optimiert werden, sondern auch die Zielfunktion gewechselt/angepasst wird (Sharpe, Calmar, Profit Factor, Utility-Varianten, Drawdown-Constraints), erweitert sich der Suchraum um eine Menge von Zielfunktionen $\mathcal{J}$:

$$K_{\text{obj}} \approx |\mathcal{J}|. \quad (17)$$

Wenn in s adaptiven Schritten (implizit) Zielfunktionsentscheidungen getroffen werden, kann eine grobe Obergrenze für die Prozedurenanzahl sein:

$$K_{\text{proc}} \sim |\mathcal{J}|^s. \quad (18)$$

Landau.

$$\ln K_{\text{proc}} = \ln(|\mathcal{J}|^s) = s \ln |\mathcal{J}| = \Theta(s).$$

Also:

$$\text{Bias} = \Theta\left(\sqrt{\frac{s}{T}}\right) \quad (\text{bei konstantem } |\mathcal{J}|). \quad (19)$$

Interpretation. Die Anzahl adaptiver Entscheidungen s wird zu einem linearen Zähler in $\ln K$. Dies ist besonders gefährlich, weil diese Entscheidungen oft *nicht* als Tests mitgezählt werden, den Suchraum aber faktisch stark vergrößern.

4.5 Baustein E: Adaptive Suchraum-Erweiterung (Space Creep) in Iterationen

Beschreibung. Ein iterativer Prozess fügt in jeder Iteration neue Freiheitsgrade hinzu (neue Indikatoren, neue Bins, neue Filterklassen). Wenn pro Iteration R neue binäre Schalter/Items hinzukommen, dann wächst die Anzahl kombinierbarer Elemente linear in R :

$$p(R) = p_0 + cR. \quad (20)$$

Wenn danach freie Subsets erlaubt sind:

$$K(R) \sim 2^{p(R)} = 2^{p_0 + cR}. \quad (21)$$

Landau.

$$\ln K(R) = (p_0 + cR) \ln 2 = \Theta(R),$$

und damit:

$$\text{Bias}(R) = \Theta\left(\sqrt{\frac{R}{T}}\right). \quad (22)$$

Interpretation. Hier ändert sich die Wachstumsordnung in der Iterationszahl R qualitativ: während reine Trial-Erhöhung $K \sim R$ nur $\sqrt{\ln R}$ -Wachstum in $\ln K$ erzeugt, führt das Hinzufügen neuer kombinierbarer Freiheitsgrade zu $\ln K = \Theta(R)$ und damit $\text{Bias} \sim \sqrt{R}$.

5 Kombination mehrerer Eskalationsmechanismen: warum es so schnell schlimm wird

5.1 Produktregel und Log-Summe

Kombiniert man mehrere Mechanismen, dann ist typischerweise:

$$K_{\text{all}} \approx \prod_{i=1}^m K_i \quad \Rightarrow \quad \ln K_{\text{all}} = \sum_{i=1}^m \ln K_i. \quad (23)$$

Setzt man (23) in (6) ein, ergibt sich:

$$\text{Bias} \approx \Theta \left(\sqrt{\frac{\sum_i \ln K_i}{T}} \right). \quad (24)$$

Kernergebnis. Wenn *mehrere* K_i kombinatorisch sind, z. B.

$$K_{\text{feat}} = 2^p, \quad K_{\text{filter+thr}} = (1 + L)^m,$$

dann ist:

$$\ln K_{\text{all}} = \Theta(p + m \ln(1 + L)), \quad (25)$$

und damit:

$$\text{Bias} = \Theta \left(\sqrt{\frac{p + m \ln(1 + L)}{T}} \right). \quad (26)$$

5.2 Beispiel 1: Feature-Subsets $\times$ Filter-Schwellen

Suchraum.

$$K = 2^p \cdot (1 + L)^m.$$

Logarithmus.

$$\ln K = \Theta(p) + \Theta(m \ln(1 + L)).$$

Bias.

$$\text{Bias} = \Theta \left(\sqrt{\frac{p + m \ln(1 + L)}{T}} \right).$$

Warum das gefährlich ist. Bereits moderate Erhöhungen in p und m erzeugen additive lineare Terme in $\ln K$ und damit einen deutlich wachsenden Bias in den Freiheitsgraden.

5.3 Beispiel 2: Verschachtelung (Outer Feature-Selection, Inner Rule-Mining)

Eine in der Praxis besonders toxische Struktur ist “außen” freie Feature-Selection und “innen” Rule-Mining bzw. weitere kombinatorische Suche.

Grobe Obergrenze. Außen: 2^p Feature-Subsets. Innen: pro Subset ein Regelraum bis zur gleichen Ordnung (vereinfachend) $\sim 2^{pb}$, wobei b Bins pro Feature entspricht. Dann:

$$K \lesssim 2^p \cdot 2^{pb} = 2^{p(1+b)}.$$

Damit:

$$\ln K = \Theta(p(1 + b)), \quad \text{Bias} = \Theta \left(\sqrt{\frac{p(1 + b)}{T}} \right).$$

Interpretation. Während “mehr Trials” nur $\ln K = \Theta(\ln n)$ erzeugt, führt diese Verschachtelung zu $\ln K = \Theta(p)$ bzw. $\Theta(pb)$: die Wachstumsordnung kippt qualitativ.

5.4 Beispiel 3: Iteratives Hinzufügen von Items/Filtern (Space Creep)

Wenn in jeder Iteration R neue Items/Filter hinzukommen und danach freie Kombinationen erlaubt sind, gilt $\ln K = \Theta(R)$ und Bias $\Theta(\sqrt{R/T})$. Kombiniert man dies zusätzlich mit Schwellenwertgittern oder weiteren Pipeline-Faktoren, addieren sich weitere Terme in (26).

6 Spezialfall FP-Growth/Association Rules im Trading: warum die Kombinationen dort besonders gefährlich sind

6.1 Items, Discretization und Itemset-Raum

FP-Growth arbeitet auf einer Menge von *Items*. Im Trading entstehen Items häufig durch Discretization (Binning) kontinuierlicher Features, z. B. “RSI in Bin 4”, “Return_{t-1} in Bin 2”.

Seien:

- p : Anzahl kontinuierlicher Basisfeatures,
- b : Anzahl Bins pro Feature (pro Feature entstehen $\Theta(b)$ Items).

Dann ist die Item-Anzahl grob:

$$N_{\text{items}} = \Theta(pb).$$

Der Raum möglicher Itemsets ist dann (im Worst Case):

$$K_{\text{itemsets}} \sim 2^{N_{\text{items}}} = 2^{\Theta(pb)} \Rightarrow \ln K_{\text{itemsets}} = \Theta(pb). \quad (27)$$

Damit:

$$\text{Bias} = \Theta\left(\sqrt{\frac{pb}{T}}\right). \quad (28)$$

Praktische Bedeutung. Schon ohne explizites “HPO” kann Rule-Mining implizit einen riesigen Suchraum erzeugen, weil jede potenzielle Regel einem Itemset entspricht. Der Prozess wird besonders problematisch, wenn Regeln nach Backtest-Metriken selektiert werden (“beste Regeln nach Sharpe/PnL”).

6.2 Antezedenzlänge und Kombinatorik

Die Anzahl möglicher Antezedenzmengen bis zur Länge L ist:

$$K_{\text{ant}} = \sum_{\ell=1}^L \binom{N_{\text{items}}}{\ell}.$$

Falls L nicht sehr klein gehalten wird, wächst K_{ant} extrem schnell; für $L = \Theta(N_{\text{items}})$ gilt $K_{\text{ant}} = \Theta(2^{N_{\text{items}}})$ und somit $\ln K = \Theta(N_{\text{items}}) = \Theta(pb)$.

6.3 Kombination mit nachgelagerten Filtern/Schwellen

In Trading-Workflows werden Regeln oft nachträglich gefiltert (z. B. Mindest-Confidence, Mindest-Lift, Mindest-PnL oder Mindest-Sharpe, Drawdown-Grenzen, Regime-Filter). Werden zusätzlich m Filterfamilien mit je L Schwellenwerten zugelassen, entsteht ein weiterer Suchraumfaktor:

$$K_{\text{filters}} \approx (1 + L)^m, \quad \ln K_{\text{filters}} = \Theta(m \ln(1 + L)).$$

Kombiniert mit dem Itemset-Raum ergibt sich:

$$K_{\text{all}} \approx 2^{\Theta(pb)} \cdot (1 + L)^m \cdot K_{\text{pipe}}.$$

Durch Logarithmieren folgt:

$$\ln K_{\text{all}} = \Theta(pb) + \Theta(m \ln(1 + L)) + \ln K_{\text{pipe}}.$$

Folglich skaliert der Selektionsbias als

$$\text{Bias} = \Theta \left(\sqrt{\frac{pb + m \ln(1 + L) + \ln K_{\text{pipe}}}{T}} \right).$$

Interpretation. Der dominante Term ist häufig bereits $\Theta(pb)$ aus dem Itemset-Raum (Discretization: p Basisfeatures, b Bins). Zusätzliche Filter- und Schwellenwertdimensionen erhöhen $\ln K$ um den additiven Term $\Theta(m \ln(1 + L))$. Nachgelagerte Pipeline-Entscheidungen (z. B. Portfolio-Konstruktion, Trading-Thresholds) wirken als weiterer multiplikativer Faktor K_{pipe} und addieren im Log-Raum $\ln K_{\text{pipe}}$.

7 7 Kombination mehrerer Eskalationsmechanismen: Gesamtwirkung auf den Bias

7.1 7.1 Produktregel und additive Struktur im Log-Raum

Werden mehrere Suchraum-erweiternde Mechanismen kombiniert, ergibt sich die effektive Anzahl getesteter Varianten typischerweise als Produkt:

$$K_{\text{all}} \approx \prod_{i=1}^M K_i.$$

Durch Logarithmieren folgt unmittelbar:

$$\ln K_{\text{all}} = \sum_{i=1}^M \ln K_i.$$

Setzt man dies in die Extremwert-Näherung für den Selektionsbias ein, ergibt sich:

$$\text{Bias} \approx \Theta \left(\sqrt{\frac{\sum_{i=1}^M \ln K_i}{T}} \right).$$

Zentrale Aussage. Die Eskalation entsteht nicht dadurch, dass der Bias schneller als $\sqrt{\ln K}$ in K wächst, sondern dadurch, dass sich mehrere große Terme in $\ln K$ addieren. Jeder kombinatorische Freiheitsgrad liefert einen linearen Beitrag im Log-Raum.

7.2 7.2 Beispiel: Feature-Subsets, Filter-Schwellen und Pipeline kombiniert

Angenommen:

- p Basisfeatures mit freier Subset-Auswahl: $K_{\text{feat}} = 2^p$,
- m Filterfamilien mit L Schwellen je Filter: $K_{\text{filter}} = (1 + L)^m$,
- zusätzliche Pipeline-Entscheidungen: K_{pipe} .

Dann gilt:

$$K_{\text{all}} = 2^p \cdot (1 + L)^m \cdot K_{\text{pipe}},$$

und somit:

$$\ln K_{\text{all}} = \Theta(p) + \Theta(m \ln(1 + L)) + \ln K_{\text{pipe}}.$$

Der Selektionsbias skaliert folglich als:

$$\text{Bias} = \Theta\left(\sqrt{\frac{p + m \ln(1 + L) + \ln K_{\text{pipe}}}{T}}\right).$$

Interpretation. Schon wenn zwei der Terme (p und m) moderat groß sind, dominiert ihre Summe den Logarithmus. Der Bias wird dann primär durch die Struktur des Suchraums bestimmt, nicht mehr durch die Anzahl explizit gezählter Trials.

7.3 7.3 Verschachtelte Eskalation (Outer Selection $\times$ Inner Mining)

Besonders kritisch ist eine verschachtelte Struktur, bei der eine kombinatorische Suche auf einer weiteren kombinatorischen Suche aufsetzt.

Beispiel:

- äußere Feature-Subset-Auswahl: 2^p ,
- inneres Rule-Mining (z. B. FP-Growth mit b Bins pro Feature): $\sim 2^{pb}$.

Eine grobe obere Schranke ist dann:

$$K_{\text{all}} \lesssim 2^p \cdot 2^{pb} = 2^{p(1+b)}.$$

Damit:

$$\ln K_{\text{all}} = \Theta(p(1 + b)), \quad \text{Bias} = \Theta\left(\sqrt{\frac{p(1 + b)}{T}}\right).$$

Konsequenz. Die Wachstumsordnung kippt qualitativ: Anstelle eines langsamen Wachstums in $\ln(\text{Trials})$ wächst der Bias wie die Wurzel eines linearen Freiheitsgrad-Zählers (p bzw. pb).

8 8 Numerisches Beispiel: Wie Ergebnisse vollständig unsignifikant werden

8.1 8.1 Setup

Es werde ein Backtest mit folgenden Parametern betrachtet:

- Stichprobengröße: $T = 252$ (ein Jahr täglicher Daten),
- annualisierte Volatilität: $\sigma = 20\%$,
- kein echter Edge: $\mu_{\text{true}} = 0$, $SR_{\text{true}} = 0$.

Der Research-Prozess enthält:

- $p = 30$ potenzielle Features (freie Subsets),
- $b = 3$ Bins pro Feature (FP-Growth-Items),
- $m = 5$ Filterfamilien mit $L = 5$ Schwellen,
- zusätzliche Pipeline-Varianten: $K_{\text{pipe}} = 50$.

8.2 8.2 Effektive Suchgröße

Die effektive Suchgröße ist näherungsweise:

$$K_{\text{all}} \approx 2^{pb} \cdot (1 + L)^m \cdot K_{\text{pipe}} = 2^{90} \cdot 6^5 \cdot 50.$$

Logarithmisch:

$$\ln K_{\text{all}} \approx 90 \ln 2 + 5 \ln 6 + \ln 50 \approx 62.4 + 8.96 + 3.91 \approx 75.3.$$

8.3 8.3 Erwarteter Selektionsbias

Der erwartete Sharpe-Bias beträgt:

$$\mathbb{E}[\widehat{SR}_{\text{max}}] \approx \sqrt{\frac{2 \ln K_{\text{all}}}{T}} = \sqrt{\frac{2 \cdot 75.3}{252}} \approx \sqrt{0.597} \approx 0.77.$$

8.4 8.4 Übersetzung in scheinbare Rendite

Die scheinbare annualisierte Rendite beträgt:

$$\Delta\mu \approx \sigma \cdot \mathbb{E}[\widehat{SR}_{\text{max}}] = 0.20 \cdot 0.77 \approx 15.4\% \text{ p.a.}$$

8.5 8.5 Interpretation der Signifikanz

Ein beobachteter Backtest mit:

$$\widehat{SR} \approx 0.7 \quad \text{und} \quad \hat{\mu} \approx 14\% \text{ p.a.}$$

würde in vielen Kontexten als *hochsignifikant* interpretiert. Unter Berücksichtigung des Selektionsbias ist dieses Ergebnis jedoch vollständig kompatibel mit der Nullhypothese $\mu_{\text{true}} = 0$.

Formal gilt:

$$\widehat{SR} = SR_{\text{true}} + \text{Bias} + \text{Schätzrauschen},$$

und der Bias-Term allein liegt bereits in der Größenordnung des beobachteten Wertes. Das Ergebnis ist daher statistisch nicht belastbar; die scheinbare Signifikanz ist vollständig durch Suchraum-Eskalation erklärbar.

9 9 Kernaussage

Wenn mehrere suchraumerweiternde Mechanismen kombiniert werden (Feature-Subsets, Rule-Mining, Filter-Schwellen, Pipeline-Produkte), wird die effektive Anzahl getesteter Varianten so groß, dass der erwartete Selektionsbias realistische Performance-Kennzahlen vollständig überdecken kann. In diesem Regime verlieren klassische Signifikanzintuitionen ihre Aussagekraft: selbst sehr hohe Sharpe-Ratios können rein durch Data Snooping und Multiple Testing entstehen.