

Makroökonomie I: Projektarbeit zur makroökonomischen Erklärung des Ölpreises

Bernd Prostmaier (Matrikelnummer: 12004113)

09. August 2021

Contents

1	Einführung in die Projektarbeit	3
1.1	FRED_MD Datensatz	3
1.2	Datenaufbereitung	3
1.3	Motivation der Projektarbeit	3
2	Deskriptive Statistiken	3
3	Bayesianische Analyse	5
3.1	Bayesianische Theorie	5
3.2	Stochastic Search Variable Selection (SSVS)	6
3.3	Modellimplementierung in R	8
3.4	Ergebnisse	13
3.4.1	Hyperparameter: $\tau_0 = 0.01, \tau_1 = 10$	13
3.4.2	Hyperparameter: $\tau_0 = 0.000001, \tau_1 = 10$	18
3.4.3	Hyperparameter: $\tau_0 = 50, \tau_1 = 10$	21
3.4.4	Hyperparameter: $\tau_0 = 0.1, \tau_1 = 100$	23
3.4.5	Special Topic: Modell ab 2000er Jahre	30
4	Conclusion	31
5	Reference	32

ABSTRACT

Die vorliegende Projektarbeit beschäftigt sich mit einem bayesianischen Modell zur Schätzung der Ölpreisenentwicklung. Insgesamt stehen hierfür acht erklärende Variablen zur Auswahl. Der verwendete Datensatz wird von der Federal Reserve Bank of St. Louis bereit gestellt und liefert Time Series Data rückwirkend bis zum 01. Januar 1959.

Das Term Paper versucht mit Hilfe von Stochastic Search Variable Selection (SSVS) das ‘beste’ Modell aus einer Reihe potenziell plausibler Modelle zu finden. Hierfür fanden Analysen für unterschiedliche Hyperparameter statt. Konkret wurden die Werte für τ_0 und τ_1 variiert, um das veränderte Verhalten auf die Schätzung zu beobachten.

Es konnte gesehen werden, dass insbesondere die Wahl von τ_0 , was die Prior Varianz für den Fall $\beta = 0$ darstellt, elementar wichtig für die Güte des Modells ist. Wählt man τ_0 zu klein, läuft man Gefahr in einen sog. absorbing state zu verfallen, wird τ_0 hingegen zu groß gewählt besteht die Gefahr, dass kein effektives ‘shrinken’ der Koeffizienten möglich ist. Während George und McCulloch in ihrem Paper einen sog. ‘semiautomatic approach’ beschreiben um geeignete Werte für Tau zu finden, konzentriert sich das Term Paper auf die Visualisation der Auswirkungen veränderter Tau-Werte.

1 Einführung in die Projektarbeit

1.1 FRED_MD Datensatz

Der vorliegende Datensatz, welcher von der Federal Reserve Bank of St. Louis veröffentlicht wird, liefert 129 makroökonomische monatliche Indikatoren in einem Zeitraum rückwirkend bis 01. Januar 1959. Die Daten werden in regelmäßigen Intervallen aus unterschiedlichen Quellen (z.B. GSI, Shiller's, etc.) zusammengetragen. Das Ziel des FRED Datensatzes (Federal Reserve Economic Data) ist, den Aufwand für die makroökonomische Analyse zu reduzieren. Die Arbeit mit einer standardisierten Datenbank soll die Replikation und den Vergleich von Ergebnissen erleichtern. Aktualisierte Formen des Datensatzes können forlaufend unter **diesem Link** abgerufen werden.

Eine detaillierte Dokumentation über den verwendeten Datensatz ist dem Paper **FRED-MD: A Monthly Database for Macroeconomic Research** aus dem Journal of Business & Economic Statistics (2015) zu entnehmen.

1.2 Datenaufbereitung

Zum Zeitpunkt dieser Arbeit wurde der Datensatz vom 30. Juni 2021 verwendet. Aus Gründen der besseren Nachvollziehbarkeit, wurde der Datensatz zunächst auf die nachfolgenden Variablen eingeschränkt:

Variable	Erklärung
OILPRICE _x	Crude Oil, spliced WTI and Cushing
CPIAUCSL	CPI : All Items
CUSR0000SAC	CPI : Commodities
S.P.500	S&P's Common Stock Price Index: Industrials
IPFUELS	IP: Fuels
IPMAT	IP: Materials
CMRMTSPL _x	Real Manu. and Trade Industries Sales
TWEXAFEGSMTH _x	Trade Weighted U.S. Dollar Index
EXCAUS _x	Canada / U.S. Foreign Exchange Rate
SASDATE	Date

Die Auflistung aller im ursprünglichen Datensatz enthaltener Variablen sind dem **Appendix** des o.g. Papers zu entnehmen.

1.3 Motivation der Projektarbeit

Nachfolgend soll die Entwicklung des Ölpreises mit Hilfe eines Bayesianischen Modells geschätzt werden. Grundlage für die Variablenselektion des Modells ist die Stochastic Search Variable Selection (SSVS). Hierbei sollen die Auswirkungen unterschiedlicher Hyperparameter auf das Modell untersucht werden.

2 Deskriptive Statistiken

Um ein Gespür für den Datensatz zu bekommen, erfolgen vor der bayesianischen Analyse verschiedene deskriptive Auswertungen.

Die Natur der Preisentwicklung am Ölmarkt besteht im Prinzip aus zwei Gruppen. Einerseits treten Produzenten und Verbraucher am Markt auf, die mit physischem Öl arbeiten und über den Börsenhandel Preise absichern um Risiken zu minimieren. Hierbei handelt es sich um Hedger, wozu beispielsweise

Ölgesellschaften oder -händler gehören, deren Business Modell die Belieferung des Marktes mit Öl oder Ölprodukten ist. Ebenso klassische Unternehmen wie Fluggesellschaften oder Speditionen, die kalkulierbare Rohstoffpreise benötigen und Preisschwankungen absichern wollen, zählen zu dieser Gruppe. Formen des Handels können in diesem Fall u.a. Termin- und Optionshandel sein. Klassischerweise findet der Terminhandel OTC (over the counter), also direkt zwischen zwei Vertragspartnern, statt. Andererseits besteht der Markt aus institutionellen und privaten Anlegern, die profitable Anlagemöglichkeiten gegen Übernahme von Risiken suchen. Diese Art von Akteuren sind jedoch meist nicht an der tatsächlichen Lieferung der Ware interessiert (kein physischer Handel). Da wie am Aktienmarkt eine breite Gruppe von Akteuren im Markt auftritt, zeigt Bild 1 eine volatile Preisentwicklung des Rohölpreises in den vergangenen 50 Jahren.

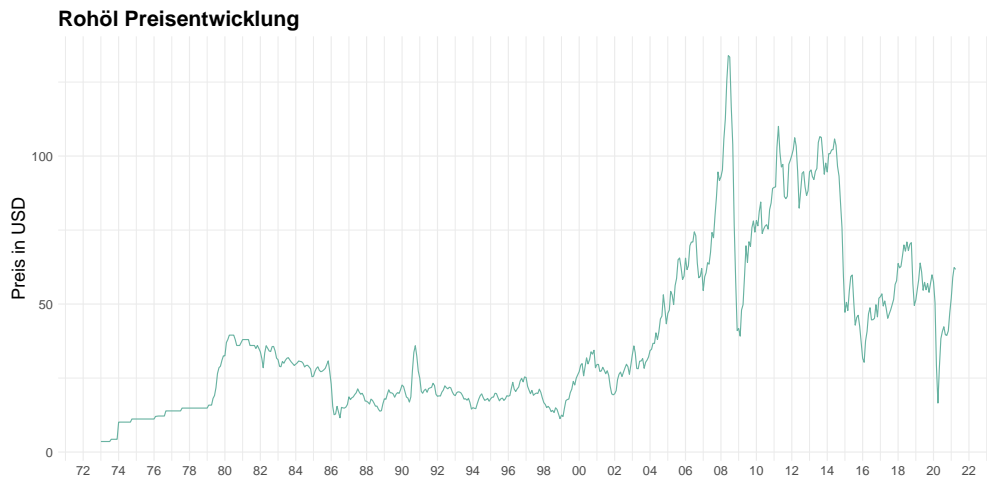


Figure 1: Ölpreisentwicklung 1972 - 2022

Der Ölpreis ist das auf einem Markt festgestellte Austauschverhältnis für eine bestimmte Menge Öl (meist ein Barrel). Bild 1 zeigt deutlich die Auswirkungen der globalen Finanzkrise 2008/2009 sowie die Auswirkungen der jüngsten Ereignisse (COVID-19 Pandemie).

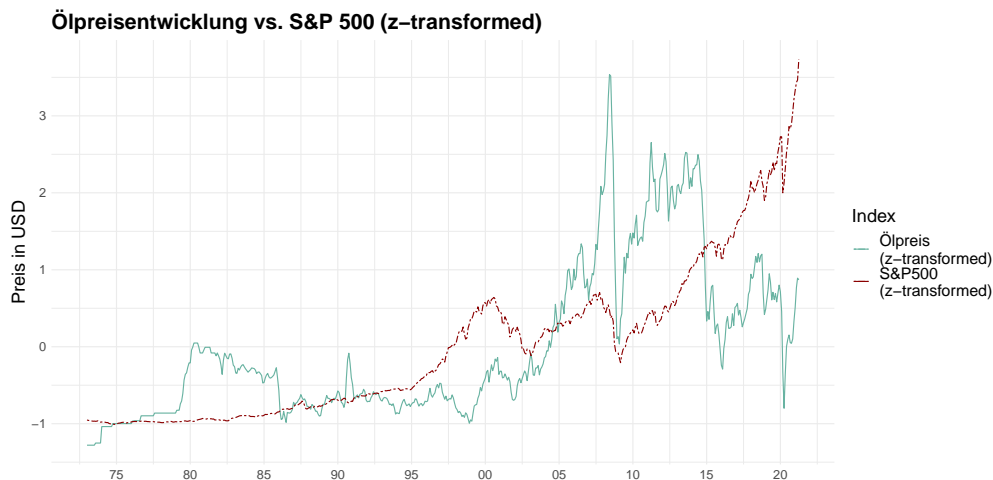


Figure 2: Ölpreisentwicklung vs. SP 500

Bild 2 zeigt die Entwicklung des S&P 500 und des Rohölpreises der vergangenen Jahre. Um beide Variablen miteinander vergleichen zu können, wurden die Variablen im Vorfeld z-transformiert. Es ist klar erkennbar, dass der Ölpreis im Vergleichszeitraum deutlich stärker schwankt als der S&P 500.

Rohöl notiert, ebenso wie viele andere international gehandelte Güter, in US Dollar. Bewegt sich der USD bei gleichbleibender Angebots- und Nachfragesituation auf dem Ölmarkt, schwankt der Rohölpreis (in US-Dollar) in etwa dem gleichen Umfang. Der Wechselkurs eines Landes gegenüber dem US Dollar ist also eine entscheidende ökonomische Größe, welche später in das bayesianische Modell einfließt. Bild 3 (links) gibt einen Überblick über die Entwicklung des Trade Weighted US Dollar Index, welcher eine Kennzahl ist, die den Wert des US Dollars mittels eines Währungskorbs aus 26 Währungen vergleicht. Der Index ist als handelsgewichteter Durchschnitt im Vergleich zu diesen Währungen zu interpretieren. Ebenso ist dem rechten Plot in Bild 3 die Entwicklung des Wechselkurses zwischen USD/CAD zu entnehmen.

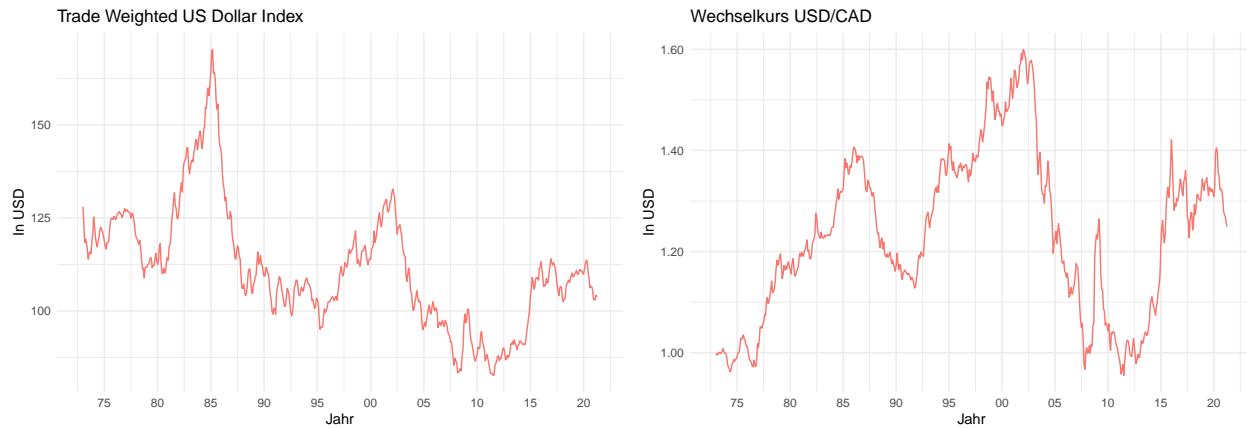


Figure 3: Währungsveränderungen

3 Bayesianische Analyse

3.1 Bayesianische Theorie

Die Grundzüge der bayesianischen Ökonometrie basieren auf den Grundlagen der Wahrscheinlichkeitsrechnung. Die Regeln der Wahrscheinlichkeitsrechnung implizieren:

$$p(A, B) = p(A|B)p(B)$$

wobei $p(A, B)$ die gemeinsame Wahrscheinlichkeit von A und B ist, $p(A|B)$ die Wahrscheinlichkeit von A gegeben B und $p(B)$ die marginale Wahrscheinlichkeit von B. Folgt man den einfachen Regeln der Wahrscheinlichkeitsrechnung folgt daraus:

$$p(A, B) = p(B|A)p(A)$$

Aus diesen Ausdrücken können wir Bayes Theorem, welches den Grundsatz für die bayesianische Ökonometrie bildet, ableiten:

$$p(B|A) = \frac{p(A|B)p(B)}{p(A)}$$

In der Ökonometrie verwenden wir Daten um ‘über etwas’ zu lernen. In diesem Beispiel sind wir an sog. ‘Parametern’ interessiert. Wenn y ein Vektor bzw. eine Matrix von Daten ist und θ ein Vektor oder eine Matrix von Parametern, für ein Modell welches y erklären soll, ist, dann sind wir am Lernen von θ , basierend auf die Daten y , interessiert. Hierfür wird Bayes Theorem verwendet:

$$p(\theta|y) = \frac{p(y|\theta)}{p(y)}$$

Aus bayesianischer Sicht sind wir grundsätzlich an $p(\theta|y)$, also der Frage - gegeben die Daten, was wissen wir über θ ? -, interessiert. Da wir nur am Lernen über θ interessiert sind, können wir $p(y)$ ignorieren, da es nicht von θ abhängt. Daraus folgt:

$$p(\theta|y) \propto p(y|\theta)p(\theta)$$

$p(\theta|y)$ wird *Posterior*, $p(y|\theta)$ wird *Likelihood* und $p(\theta)$ wird *Prior* genannt. Aus der Gleichung folgt schließlich, dass der *Posterior proportional zum Produkt der Likelihood und des Priors* ist. Wichtig ist, dass der Prior $p(\theta)$ nicht von den Daten y abhängt! Er enthält also das Wissen über θ bevor wir die Daten gesehen haben. Die Likelihood funktion $p(y|\theta)$ ist die Dichte der Daten gegeben den Parametern und wird laut Koop oft als datengenerierender Prozess bezeichnet. Der Posterior $p(\theta|y)$ gibt uns alles zurück was wir über θ wissen nachdem wir (a posteriori) die Daten gesehen haben - er kombiniert also 'data und non-data' Informationen.

Eine später im Kapitel 3.3 angewandte Technik zur Simulation des Posteriors ist das sog. 'Gibbs Sampling', welches eine sequentielle Ziehung aus den vollständig bedingten Posterior-Verteilungen ist. Während Monte Carlo Integration zufällige Ziehungen von $p(\theta|y)$ vornimmt und diese anschließend mittelt um Schätzungen für $E[g(\theta)|y]$, wenn $g(\theta)$ eine beliebige Funktion ist, zu erhalten, verfolgt Gibbs Sampling einen anderen Ansatz. Oft ist es nicht einfach aus $p(\theta|y)$ zu ziehen, aber es ist oftmals einfach zufällig von $p(\theta_1|y, \theta_2, \dots, \theta_B), p(\theta_2|y, \theta_1, \theta_3, \dots, \theta_B), \dots, p(\theta_B|y, \theta_1, \dots, \theta_{B-1})$ zu ziehen. Die vorangegangene Verteilungen werden 'full conditional posterior distributions' genannt, da sie einen Posterior für jeden Block, konditioniert auf alle anderen Blöcke, darstellen. Es kommt also zu einer Sequenz von $\theta^1, \theta^2, \dots, \theta^s$, welche gemittelt eine Schätzung $E[g(\theta|y)]$ ergeben, wie es die Monte Carlo Integration macht.

Insgesamt finden S Ziehungen statt, was θ^s für $s = 1, \dots, S$ ergibt. Nachdem die ersten Ziehungen S_0 (sog. *burn-in*) wegfallen, um den Einfluss von θ^0 zu eliminieren, werden die restlichen Ziehungen S_1 gemittelt um Schätzungen für $E[g(\theta|y)]$ zu erhalten.

$$\hat{g}_{S_1} = \frac{1}{S_1} \sum_{s=S_0+1}^S g(\theta^s)$$

Letzendlich kann gezeigt werden, dass \hat{g}_{S_1} zu $E[g(\theta|y)]$ konvergiert, wenn S_1 gegen unendlich geht!

3.2 Stochastic Search Variable Selection (SSVS)

Der eingeschränkte Datensatz enthält neun Variablen, wovon acht als erklärende Variablen dienen sollen (siehe Kapitel 1). SSVS hat die Auswahl eines Modells aus einer Reihe von potenziell plausiblen Modellen zur Aufgabe. Es soll also die Auswahl einer optimalen Teilmenge von Variablen, aus der Menge aller verfügbaren Kovariaten, gefunden werden. Insgesamt stehen bei der Regressionsanalyse 2^p Modelle, wenn p Prädiktoren zur Verfügung stehen, unter der Annahme, dass der Achsenabschnitt eingeschlossen ist und keine Wechselwirkungen zwischen den Kovariaten bestehen, zur Auswahl. Der Modellraum kann daher schnell extrem groß sein. Dies kann dazu führen, dass die vollständige Aufzählung aller potenzieller Modelle und die Bewertung der Posterior Wahrscheinlichkeiten extrem erschwert wird. Aus diesem Grund wird im Nachfolgenden ein SSVS Prior verwendet um den Modellraum effizient zu erforschen.

Hierfür wird ein binärer Vektor eingeführt: γ . Dieser wird verwendet, um anzugeben, ob eine Variable in das Modell aufgenommen oder aus ihm ausgeschlossen wird (aktiv oder inaktiv). Weiterhin sorgt das SSVS Modell dafür, dass jeder Regressionskoeffizient nicht genau gleich Null gesetzt wird, wenn eine Kovariate als inaktiv aufgenommen wird, sondern a posteriori auf eine kleine Nachbarschaft um Null herum gesetzt wird.

Der binäre γ Vektor ermöglicht dem Gibbs Algorithmus die Implementierung lokaler Änderungen an einzelnen Kovariaten bei der Durchsuchung des Modellraums. Mit Hilfe von SSVS können wir also Gibbs-Sampling

verwenden, um große Modellräume zu untersuchen und posterior Wahrscheinlichkeiten direkt zu schätzen, ohne die marginale Wahrscheinlichkeit jedes Modells zu bewerten.

Beispiel:

Im linearen Regressionsmodell nehmen wir an, dass n Observationen einer abhängigen Variable Y bestehen. Weiterhin ist ein Set von Kovariaten X_1, X_2, \dots, X_p gegeben und wir nehmen an, dass

$$Y = \mathcal{N}_n(X\beta, \sigma^2 I)$$

wobei Y die Dimension $n \times 1$ hat, $X = [X_1, X_2, \dots, X_p]$ die $n \times p$ Designmatrix, $\beta = (\beta_1, \beta_2, \dots, \beta_p)^T$ der Vektor der Regressionsparameter und σ^2 ein Skalar ist. Die Auswahl einer Teilmenge von Predictoren X ist also gleichbedeutend mit dem Nullsetzen der Beta Werte der nicht im Modell enthaltenen Predictoren. Ein Hauptmerkmal der SSVS Methode ist demnach, dass X alle möglichen Prädiktoren enthält und β eine feste Dimensionalität p , für alle 2^p Modelle, hat. Wie bereits erwähnt, wird kein Parameter genau auf 0 gesetzt. Dies wird erreicht, indem ein Mix aus Normalverteilungen als Priorverteilungen für jeden Modellkoeffizienten β_j verwendet wird. Der Prior für jedes β_j , gegeben die latente Gamma Variable γ_j , ist gegeben durch:

$$\beta_j | \gamma_j \sim (1 - \gamma_j) \mathcal{N}(0, \tau_j^2) + \gamma_j \mathcal{N}(0, \tau_j^2 c_j^2)$$

und

$$P(\gamma_j = 1) = 1 - P(\gamma_j = 0) = \pi_j$$

für $j = 1, 2, \dots, p$. Der Grund für die o.g. Prior Formulierung ist, dass τ_j^2 klein gesetzt und $\tau_j^2 c_j^2$ vergleichsweise groß gesetzt werden kann. Wenn also $\gamma_j = 1$ ist, ist β_j mit einer ‘unsicheren’ Prior Distribution im Modell, sodass die posterior Verteilung hauptsächlich auf den Daten beruht. Ist im Gegensatz $\gamma_j = 0$, haben wir eine hohe Präzision (z.b. $\beta_j | \gamma_j \sim \mathcal{N}(0, 0.0001)$) - somit wird der Parameter Richtung Null geschrumpft.

Um das SSVS Modell komplett durchzumodellieren, muss ebenso noch eine Prior Verteilung für σ^2 und γ angenommen werden. Für σ^2 bietet sich beispielsweise eine conjugate Invers-Gamma Prior Verteilung an:

$$\sigma^2 | \gamma \sim IG(v_\gamma/2, v_\gamma \lambda_\gamma/2)$$

Wie im Paper von George und McCulloch beschrieben, bietet die Konditionierung auf γ Flexibilität, im Sinne der Berücksichtigung von Abhängigkeiten. Wie in der Vorlesung gesehen (und im Paper von George und McCulloch ebenfalls vorgeschlagen) gilt:

$$\pi(\gamma) = \prod_{j=1}^p \pi_j^{\gamma_j} (1 - \pi_j)^{(1-\gamma_j)}$$

Der Prior in zuletzt genannter Gleichung impliziert, dass das Inkludieren von X_z unabhängig von der Inklusion von X_j , für alle $z \neq j$, ist. Gegeben den o.g. Gleichungen, sind die bedingten Wahrscheinlichkeiten von β (Normal), σ^2 (Inverse-Gamma) und γ (Bernoulli) bekannt. Dies führt laut George und McCulloch zu einem schnellen und effizienten Gibbs sampling.

Der Output des SSVS ist ein Posterior Sample $\{\beta^t, \sigma^t, \gamma^t\}$ für $t = 1, \dots, T$. Schwerpunkt des SSVS liegt auf γ , da die relativen Häufigkeiten aller gesampelten Kombinationen von γ , Schätzungen für die ‘posterior model probabilities’ ergeben. Die sog. ‘Posterior Inclusion Probability’ (nachfolgend PIB, mehr Informationen hierzu in Kapitel 3.4) jeder Kovariate X_j kann anschließend einfach berechnet werden durch:

$$\hat{P}(\gamma_j = 1 | Y) = T^{-1} \sum_{t=1}^T \gamma_j^t$$

Weiterhin ist die BMA Schätzung, für jede Größe von Interesse (θ), einfach der Sample Mittelwert $\bar{\theta}$, den wir aus dem SSVS Output generieren, da θ^t bei jeder Iteration t von der ‘full conditional posterior distribution’ des Modells γ^t generiert wird.

3.3 Modellimplementierung in R

Zur Implementierung des vorangegangenen theoretischen Teils, dient nachfolgender R-Code (siehe Kommentare). Dieser dient hauptsächlich der grafischen Interpretation der Ergebnisse aus dem bayesianischen Modell und stellt den Schwerpunkt dieses Term Papers dar. Der Prior auf Sigma wird hierfür standardmäßig uninformativ gesetzt. Ebenso wird die Feature Matrix, neben den acht Kovariaten aus Kapitel 1, um zwei normalverteilte Zufallsvariableln (x1 und x2) ergänzt. Dies dient zur Kontrolle der Variablenselektion aus dem SSVS Modell.

Der numerische Output am Ende jeder Analyse wird nachfolgend, aufgrund der bereits aufbereiteten Form, mit Hilfe der von Valverde, Losert, Marsoner und mir gecodeten R6 Klasse dargestellt.

```
SSVS = function(X, y, nsave = 1000, nburn = 1000, tau0 = 0.01,
               tau1 = 10, S0 = 0.01, s0 = 0.01, scale = FALSE){

  # Modell preliminaries
  X = as.matrix(X)
  N = nrow(X)

  ## Gibbs preliminaries
  nsave = nsave           # S_1
  nburn = nburn           # S_0
  ntot = nsave+nburn      # S

  ## Prior preliminaries für SSVS
  tau0 = tau0             # Prior Varianz für den Fall Beta_j == 0
  # -> Tau0 muss klein sein, damit die Präzision hoch ist
  # (siehe Kapitel 3.2)
  tau1 = tau1             # Prior Varianz für den Fall Beta_j != 0

  ## Prior auf Sigma
  s0 = s0 # uninformativ
  S0 = S0 # uninformativ

  ## Konstruieren einer neuen Designmatrix durch Hinzufügen
  ## nicht relevanter Kovariaten
  Y = matrix(y)
  X = cbind(rnorm(N,0,10),rnorm(N,0,1), X) # Designmatrix

  if (scale == TRUE){
    Y = scale(Y)
    X = scale(X)
  }

  colnames(X)[1] = "x1"
  colnames(X)[2] = "x2"
```



```

N = nrow(Y)
K = ncol(X)

## Berechnung der OLS Größen
A.OLS = solve(crossprod(X))%*%crossprod(X,Y)
SSE = crossprod(Y-X)%*%A.OLS
SIG.OLS = SSE/(N-K)

## Gamma
## Startwert für Gamma wird festgelegt
## gamma == gamma_1, gamma_2, ..., gamma_K (siehe Kapitel 3.2)
## Gamma == 0 -> Variable nicht inkludiert
## Gamma == 1 -> Variable inkludiert

gamma = matrix(1,K,1)
# -> Startwert von Gamma ist ein Einheitsvektor. D.h. es wird mit dem
# vollen Modell gestartet. Zu Beginn existiert also die Annahme, dass alle
# Variablen für das Modell wichtig sind.

## Startwert für Sigma
sigma2.draw = as.numeric(SIG.OLS) # Start mit OLS Schätzer

## Damit Gibbs verwendet werden kann, muss die Prior Varianz angepasst werden
V.prior = diag(as.numeric(gamma*tau1+(1-gamma)*tau0))
# -> Abhängig von gamma können wir also jetzt flexibel bestimmen, ob die Varianz
# im Prior groß oder klein ist. Ist gamma == 0, wäre auch V.prior nahe 0. Ist
# gamma == 1, wäre V.prior abhängig von tau_1 z.B. 100 oder 1000,
# also sehr hoch.

## Storage matrizen
ALPHA.store = matrix(NA,nsave,K) # Regressionskoeffizienten
SIGMA.store = matrix(NA,nsave,1) # Fehlervarianz
Gamma.store = matrix(NA,nsave,K) # Gamma

## Gibbs Sampler
for (irep in 1:ntot){
  # Draw ALPHA given rest from multivariate normal
  # Ziehen von den Regressionskoeffizienten
  V.post = solve(crossprod(X)*1/sigma2.draw+diag(1/diag(V.prior)))
  A.post = V.post%*%(crossprod(X,Y)*1/sigma2.draw)
  A.draw = A.post+t(chol(V.post))%*%rnorm(K)

  # Draw indicators conditional on ALPHA
  # Hierbei wird jeder Koeffizient getestet
  # Wir schauen für jeden Koeffizienten, ob wir eine große oder kleine
  # Varianz im Prior brauchen!
  for (jj in 1:K){

```

```

p0 = dnorm(A.draw[[jj]],0,sqrt(tau0))
p1 = dnorm(A.draw[[jj]],0,sqrt(tau1))
p11 = p1/(p0+p1)
# -> Posterior Wahrscheinlichkeit, dass gamma_j == 1 gilt

if (p11>runif(1)) gamma[[jj]] = 1 else gamma[[jj]] = 0
# -> Vergleich p11 mit Einheitsverteilung
# -> If TRUE: gamma_j == 1, ELSE: gamma_j == 0
}

# Construct prior VC matrix conditional on gamma
# Neuer V.prior auf Basis der aktualisierten gamma_j's
V.prior = diag(as.numeric(gamma*tau1+(1-gamma)*tau0))

# Simulate sigma2 from inverse Gamma
S.post = crossprod(Y-X%*%A.draw)/2+S0
s.post = S0+N/2
sigma2.draw = 1/rgamma(1,s.post,S.post)

# Speichern der Größen nach burn-in
if (irep>nburn){
  ALPHA.store[irep-nburn,] = A.draw
  SIGMA.store[irep-nburn,] = sigma2.draw
  Gamma.store[irep-nburn,] = gamma
}
}

## Posterior Inclusion Probabilities
## Marginale Posterior Wahrscheinlichkeit, dass eine Variable im Modell
## inkludiert ist!
PIP.mean = apply(Gamma.store,2,mean)
# -> PIP ist nichts anderes, als der Mittelwert über Gamma.store

# Interpretation:
# in x.% der Fälle ist variable y inkludiert

A.mean = apply(ALPHA.store,2,mean)
SIG.mean = apply(SIGMA.store,2,mean)

##### PLOTS #####

### PIP Plot
colnames(X)[1] = "x1"
colnames(X)[2] = "x2"

PIPs = as.data.frame(PIP.mean)
PIPs$predictor = colnames(X)
PIPs$predictor = factor(PIPs$predictor, levels = PIPs$predictor)

PIP_Plot = ggplot(PIPs, aes(x=predictor, y=PIP.mean)) +

```

```

geom_segment(
  aes(x=predictor, xend=predictor, y=0, yend=PIP.mean),
  color=ifelse(PIPs$PIP.mean > 0.5, "orange", "grey"),
  size=ifelse(PIPs$PIP.mean > 0.5, 1.3, 0.7)
) +
geom_point(
  color=ifelse(PIPs$PIP.mean > 0.5, "orange", "grey"),
  size=ifelse(PIPs$PIP.mean > 0.5, 5, 2)
) +
theme_light() +
theme(
  panel.grid.major.x = element_blank(),
  panel.border = element_blank(),
  axis.ticks.x = element_blank()
) +
ylim(c(0,1)) +
labs(title = "Posterior Inclusion Probability") +
theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust=1)) +
theme(plot.title = element_text(size = 20, face = "bold"),
  axis.title.x = element_text(size = 17),
  axis.title.y = element_text(size = 17),
  legend.title=element_text(size=17),
  legend.text=element_text(size=15),
  text = element_text(size = 15))
PIP_Plot

### Alpha_store density für alle Koeffizienten bei denen PIP.mean > 0.5 ist
## Base R
plot(density(ALPHA.store[,1]), xlim = c(-1, 3), col = "grey",
  main = "Density Plot", bty = "n", xlab = "")
grid(NA, NULL, lty=3, lwd=1, col="lightgrey")
abline(v=c(-1,0,1,2,3), lty=3, lwd=1, col="lightgrey")

for (i in 1:10){
  if (PIPs$PIP.mean[i] > 0.5){
    lines(density(ALPHA.store[,i]), col = adjustcolor('orange', blue.f = 0.8), lwd = 2)
    text(mean(ALPHA.store[,i]), 4, colnames(X)[i], srt = -30, cex = 0.7)
  } else {
    lines(density(ALPHA.store[,i]), col = adjustcolor('grey', alpha.f = 0.5))
  }
}
abline(v = c(0), lty = 2, lwd = 1, col = "firebrick")

PIP_Density_Plot = recordPlot()

### All Density Plot
## GGLOT Data
Alphas = as.data.frame(ALPHA.store)
colnames(Alphas) = colnames(X)

## Get GGPlot Format

```

```

Alphas2 = stack(Alphas)

# Density Plot
All_Densities = ggplot(Alphas2, aes(x = values, y = ind, fill = ind)) +
  geom_density_ridges() +
  theme_ridges() +
  theme(legend.position = "none") +
  labs(title = "Density plot", y = "Covariate", x = "")

## Plot2
# Density excl ESCAUSx Plot
Alphas3 = Alphas2[!Alphas2$ind == "EXCAUSx", ]

Ex_Densities = ggplot(Alphas3, aes(x = values, y = ind, fill = ind)) +
  geom_density_ridges() +
  theme_ridges() +
  theme(legend.position = "none") +
  labs(title = "Density plot", y = "Covariate", x = "")

##### Ergebnisse plotten #####

beta.mean = apply(ALPHA.store, 2, mean)

# Ts.plot
Estimation = ggplot(FRED2, aes(x = sasdate)) +
  geom_line(aes(y = Y, color = "Ölpreis Data")) +
  geom_line(aes(y = as.numeric(X %*% beta.mean), color = "Posterior Model Estimation")) +

  labs(
    title = "Ölpreisentwicklung vs. Bayesianische Schätzung",
    y = "Preis in USD",
    x = "",
    colour = "Legende"
  ) +
  scale_color_manual(values = c("lightgrey", "orange")) +
  theme_minimal() +
  theme(
    legend.position = c(0.15, 0.9),
    legend.background = element_rect(linetype = "solid")
  )

if (scale == TRUE){
  Estimation = Estimation + labs(
    title = "Ölpreisentwicklung vs. Bayesianische Schätzung (skalierte Werte)",
    y = "Preis in USD (skaliert)"
  )
}

### FIT
pred = as.numeric(X%*%beta.mean)

```

```

results = as.data.frame(cbind(pred, Y))
colnames(results) = c("Prediction", "Y")
results$MSE = (results$Y - results$Prediction)^2 / N

MSE = colSums(results)$MSE

### Return
ssvs = list(PIP_Plot, All_Densities, Ex_Densities, Estimation, MSE, results)
#names(ssvs) = c("PIP Übersicht", "Density Covariate", "All Density", "Ex Density", "Results")
}

```

3.4 Ergebnisse

Wir können nun unter Angabe der Hyperparameter die Funktion aus Kapitel 3.3 ausführen. Nachfolgend finden unterschiedliche Tests mit Hilfe verschiedener Hyperparameter statt. Hierbei soll τ_0 sowie τ_1 variiert werden, um ihren Effekt auf die bayesianische Schätzung zu beobachten.

Grundsätzlich gilt, dass τ_0 eher einen kleinen Wert annehmen muss, damit die Präzision hoch ist. Wenn der Beta Koeffizient geschränkt wird, somit also Variablenselektion betrieben werden soll, muss die Präzision hoch sein, da τ_0 die Prior Varianz für den Fall $\beta_j = 0$ ist.

3.4.1 Hyperparameter: $\tau_0 = 0.01, \tau_1 = 10$

```

modell1 = SSVS(X = FRED2[, 2:9], y = FRED2$OILPRICE, tau0 = 0.01, tau1 = 10)

```

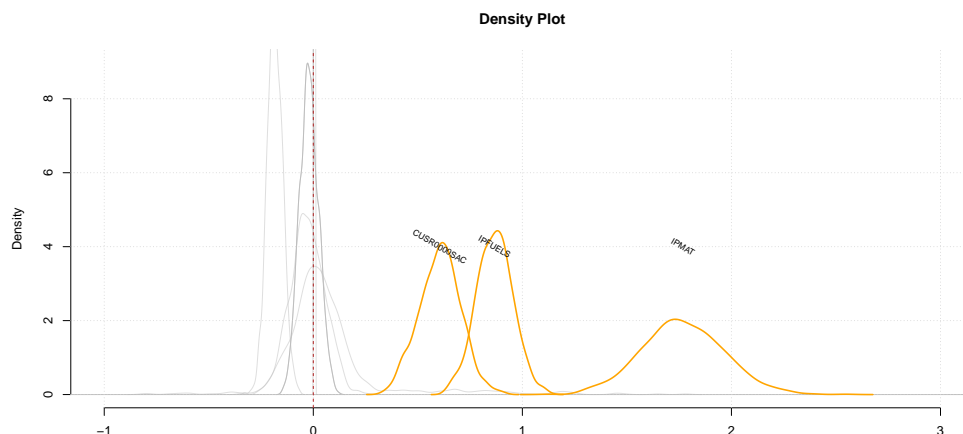


Figure 4: Density Plot ($\tau_0 = 0.01, \tau_1 = 10$)

Das Ausführen der Funktion liefert uns einen ersten Überblick über die Dichte der Koeffizientenschätzer (Bild 4). Der Plot liefert einen ersten Anhaltspunkt, welche Variablen im Modell ‘wichtig’ sind und welche nicht. Koeffizienten bei denen die Dichte in oranger Farbe angezeigt wird, haben eine Posterior Inclusion Probability (PIB) von über 50%, d.h. in über 50% der Fälle ist die jeweilige Kovariate im Modell inkludiert.

Um jedoch die genauen PIBs für jede Kovariate anzusehen, kann im Folgenden das erste Element der Funktionsliste ausgegeben werden:

```
model1[[1]]
```

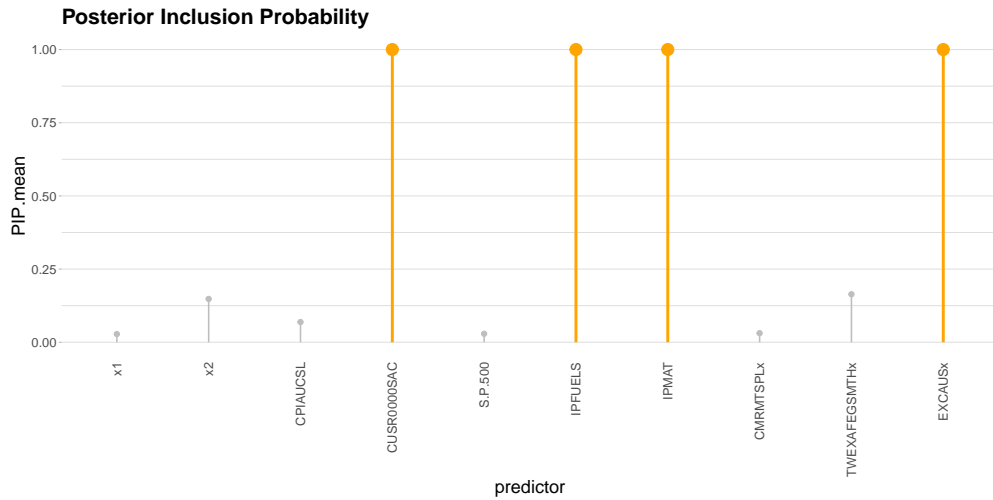


Figure 5: Posterior Inclusion Probability ($\tau_0 = 0.01, \tau_1 = 10$)

Wenn wir also τ_0 auf 0.01 setzen und τ_1 auf 10, dann erhalten wir PIBs von nahe 100% bzw. 100% für die Variablen CURSR0000SAC (CPI: Commodities), IPFUELS (IP: Fuels), IPMAT (IP: Materials) und EXCAUSx (USD/CAD Exchange Rate). Bild 5 zeigt, dass die vier genannten Kovariaten in (nahe) 100% der Fälle im Modell inkludiert sind. Da Bild 4 auf einem statischen Intervall geplottet wird, sehen wir in Bild 6 die Dichte aller Koeffizienten.

```
model1[[2]]
```

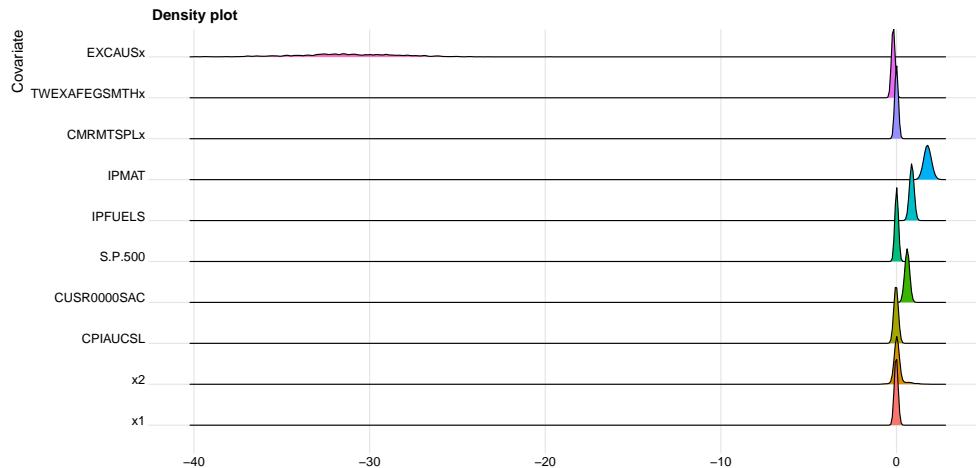


Figure 6: Dichte aller berücksichtigter Koeffizienten ($\tau_0 = 0.01, \tau_1 = 10$)

Neben den drei Dichtefunktionen aus Bild 4, sehen wir nun auch die Dichte des Koeffizienten der Variable EXCAUSx (welche zentriert um ca. -30 ist und somit in Bild 4 nicht erkennbar ist). Bild 6 zeigt somit noch einmal die grafische Aufbereitung der Dichtefunktionen der in Bild 5 dargestellten Koeffizienten. Hierbei ist klar ersichtlich, dass die Koeffizienten der Variablen IPMAT, IPFUELS, CURSR0000SAC und EXCAUSx nicht auf 0 zentriert sind, da deren latente Variable (γ_j), wie aus Bild 5 zu entnehmen ist, in (nahezu) 100%

der Fälle auf 1 gesetzt ist und die jeweilige Variable somit im Modell berücksichtigt wird. Weiterhin zeigt Bild 6 aber auch deutlich eine Schwäche auf. Die Variablen in X leben auf unterschiedlichen Skalen. Dies führt zu Problemen, da die einfache Implementierung von τ_0 und τ_1 fixe Werte vorsieht, welche unabhängig von der Skalierung der Featurematrix X sind. Um diesem Problem entgegen zu wirken, findet nachfolgend eine Skalierung von X und Y im ‘modell1’ statt:

```
modell1.2 = SSVS(X = FRED2[, 2:9], y = FRED2$OILPRICE, tau0 = 0.01, tau1 = 10, scale = TRUE)
```

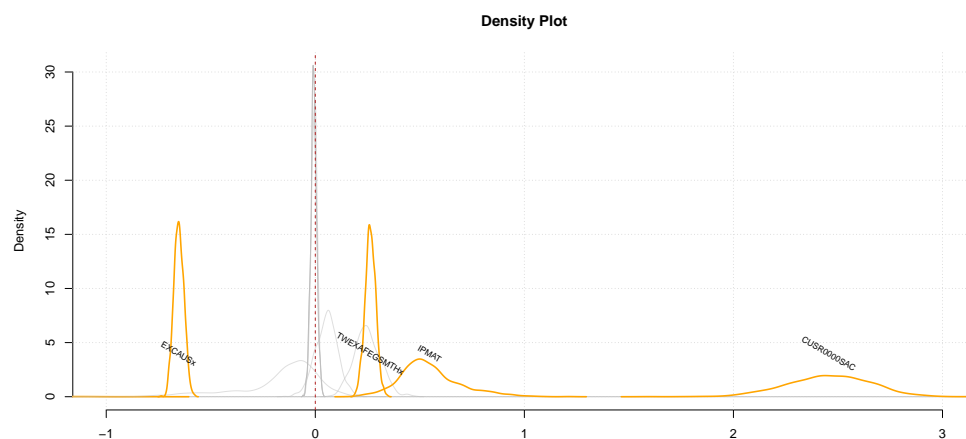


Figure 7: Density Plot ($\tau_0 = 0.01, \tau_1 = 10$) (skaliert)

Bild 7 zeigt das Ergebnis skalierten Daten. Hierbei ist bereits ersichtlich, dass sich nun Veränderungen in der Posterior Inclusion Probability ergeben haben. Bild 8 zeigt die PIBs der einzelnen Kovariaten. Während im unskalierten Modell die Variable ‘IPFUELS’ in über 50% der Modelle inkludiert war, ist diese nun in den meisten Fällen nicht mehr enthalten. Die Variable CPIAUCSL hingegen ist in 100% der Modelle enthalten. Es scheint, dass durch die Standardisierung der Featurematrix eine deutliche Verbesserung der Variablenselektion stattfindet.

```
modell1.2[[1]]
```

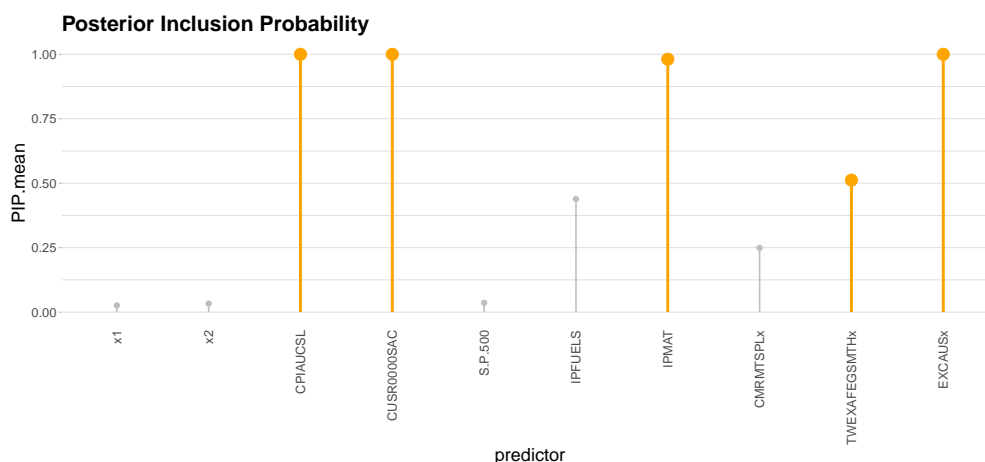


Figure 8: Posterior Inclusion Probability ($\tau_0 = 0.01, \tau_1 = 10$) (skaliert)

Bild 9 zeigt den Effekt der standardisierten Variablen am eindruckvollsten:

```
model1.2[[2]]
```

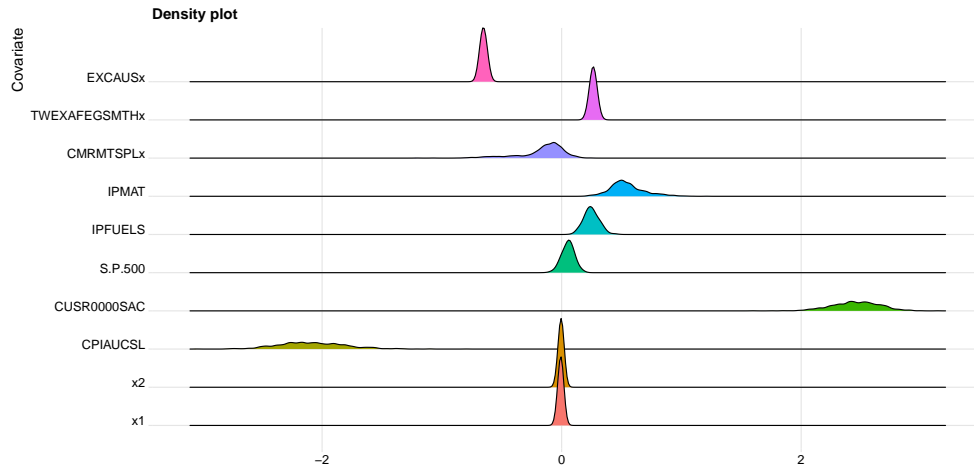


Figure 9: Dichte aller berücksichtigter Koeffizienten ($\tau_0 = 0.01, \tau_1 = 10$) (skaliert)

In Bild 10 findet nun eine Gegenüberstellung der Vorhersagen aus dem bayesianischen Modell mit den tatsächlichen Werten der abhängigen Variable (Ölpreis) statt. Hierfür befinden sich in Bild 10 zwei Plots. Links: Gegenüberstellung mit der Vorhersage aus den nicht standardisierten Daten; Rechts: Gegenüberstellung mit der Vorhersage aus den standardisierten Daten.

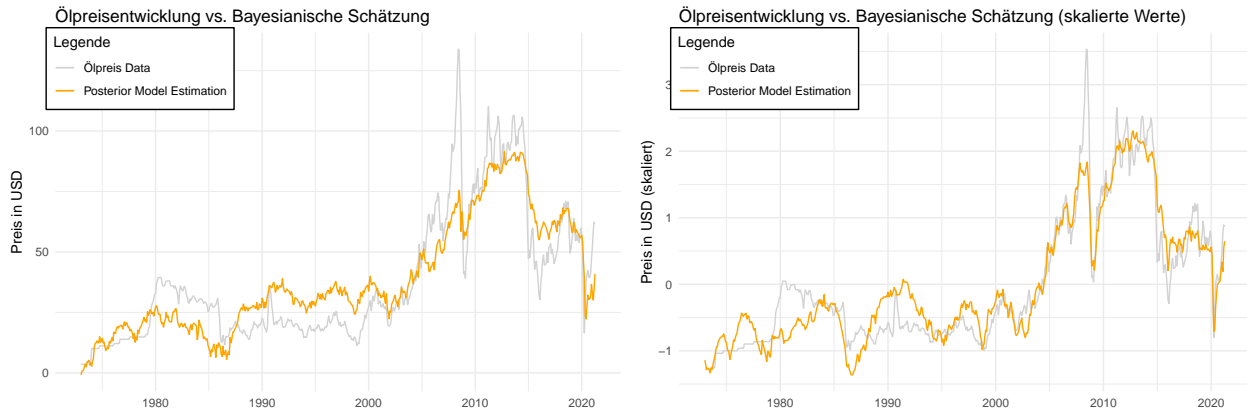


Figure 10: Daten vs. bayesianische Schätzung ($\tau_0 = 0.01, \tau_1 = 10$)

Insbesondere in Zeiten erhöhter Volatilität (z.B. Finanzkrise 2008/2009) sehen wir deutlich stabilere Ergebnisse aus dem Modell mit skalierten Daten. Da ersichtlich ist, dass ab dem Jahr 1998 im standardisierten Modell deutlich präzisere Ergebnisse erzielt werden, erfolgt nachfolgend der Vergleich unterschiedlicher Hyperparameter unter Verwendung standardisierter Werte. Mit den verwendeten Tau Werten lässt sich die abhängige Variable bereits sehr gut reproduzieren. Als Maß zur Güte der Schätzung, verwende ich im Nachfolgenden den Mean Squared Error

$$MSE = \frac{1}{N} \sum_{i=1}^N (y_i - f(x_i))^2$$

sowie die Veränderung des Akaike Informationskriteriums (AIC), welches der Idee von Ockhams Rasiermesser folgt und somit Modellkomplexität bestraft. Hierbei wird um die Anzahl der Parameter k bestraft.

$$AIC = 2k + n * \log(\sigma^2) - 2C = 2k + n * \log(RSS) - (n * \log(n) + 2C)$$

Da jedoch nur die Veränderung des AIC wirklich bedeutend ist, und der Ausdruck $n * \log(n) + 2C$ eine Konstante ist, da C ein fixer beliebiger Wert ist, kann diese Konstante ignoriert werden. Zum Modellvergleich kann daher nachfolgender Ausdruck herangezogen werden:

$$\Delta AIC = 2k + n * \ln(RSS)$$

Gütemaß	Wert des Informationskriteriums
MSE	0.1202708
AIC	2,472.111

Abschließend sind der von unserer Gruppe aus Hausübung 2 gecodeten R6 Klasse ergänzend die nachfolgenden Statistiken, betreffend Modell 1.2, zu entnehmen:

```
## Coefficients:
##              Probability          Mean          SD
## CPIAUCSL          1.000 -2.08982769  0.28001640
## CUSR0000SAC        1.000  2.46511309  0.19468801
## S.P.500            0.034  0.05612334  0.05265980
## IPFUELS            0.463  0.25731778  0.06176408
## IPMAT             0.986  0.55344890  0.14824753
## CMRMTSPLx         0.309 -0.18034578  0.21057647
## TWEXAFEGSMTHx     0.505  0.26213167  0.02601461
## EXCAUSx           1.000 -0.64993230  0.02394649
##
## N: 580  Burins: 1000  Draws: 2000
## Tau0:  0.01  Tau1:  10
```

3.4.2 Hyperparameter: $\tau_0 = 0.000001, \tau_1 = 10$

Nachfolgend gilt: $\tau_0 = 0.000001$! τ_0 wird also bewusst extrem klein gesetzt um die Auswirkung einer extrem hohen Präzision für den Fall $\beta_j = 0$ zu beobachten:

```
set.seed(2)
model2 = SSVS(X = FRED2[, 2:9], y = FRED2$OILPRICE, tau0 = 0.000001, tau1 = 10, scale = TRUE)
```

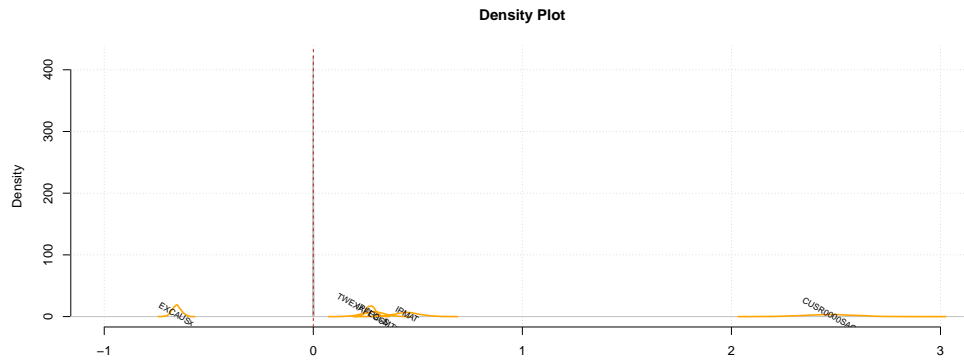


Figure 11: Density Plot $\tau_0 = 0.000001, \tau_1 = 10$

Während im Dichte Plot (Bild 11), aufgrund der fixen Achsen, kaum relevante Informationen erkennbar sind, liefert uns Bild 12 ein genaueres Bild der Dichte der Schätzkoeffizienten:

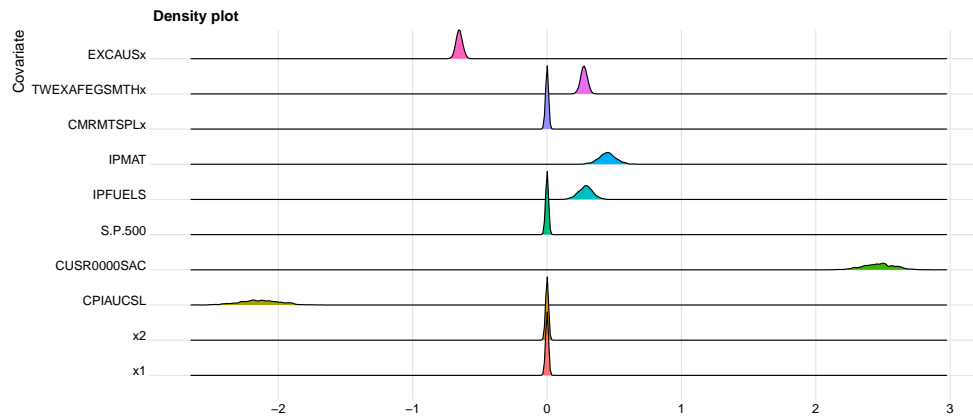


Figure 12: Dichte aller berücksichtigter Koeffizienten ($\tau_0 = 0.000001, \tau_1 = 10$)

Ein Blick auf die Posterior inclusion probabilities (Bild 13) zeigt, dass nun neben den im Modell 1 genannten Kovariaten auch noch die Variablen 'TWEXAFEGSMTHx' (Trade Weighted U.S. Dollar Index) sowie 'IPFUELS' in (nahe) 100% der Fälle enthalten sind. Die Ergebnisse aus Bild 13 sind insofern problematisch, als dass wir vermutlich verfälschte Ergebnisse erhalten, da die Wahrscheinlichkeiten strikt 0 oder 1 über alle Modelle hinweg betragen.

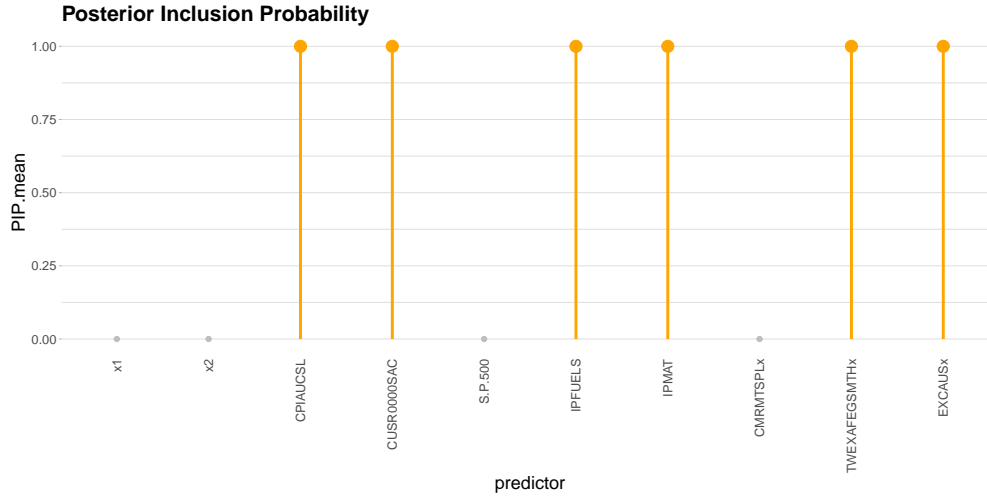


Figure 13: Posterior Inclusion Probability ($\tau_0 = 0.000001, \tau_1 = 10$)

Es scheint als wäre die verwendete Prior Varianz für den Fall $\beta_j = 0$ ($\tau_0 = 0.000001$) zu klein gesetzt. Bild 13 erweckt den Anschein, als würden wir uns in einem sog ‘absorbing state’ befinden, da die Wahrscheinlichkeiten der ‘posterior inclusion’ nurnoch 0 oder 1 betragen. Beim absorbing state handelt es sich um einen Zustand den man, wenn man ihn einmal erreicht, nicht mehr verlassen kann. Hierbei kann der Gibbs Sampler nicht mehr springen, was letztendlich dazu führt, dass wir den Koeffizienten nicht mehr shrinken sondern direkt oder sehr nahe 0 (bzw. 1) setzen.

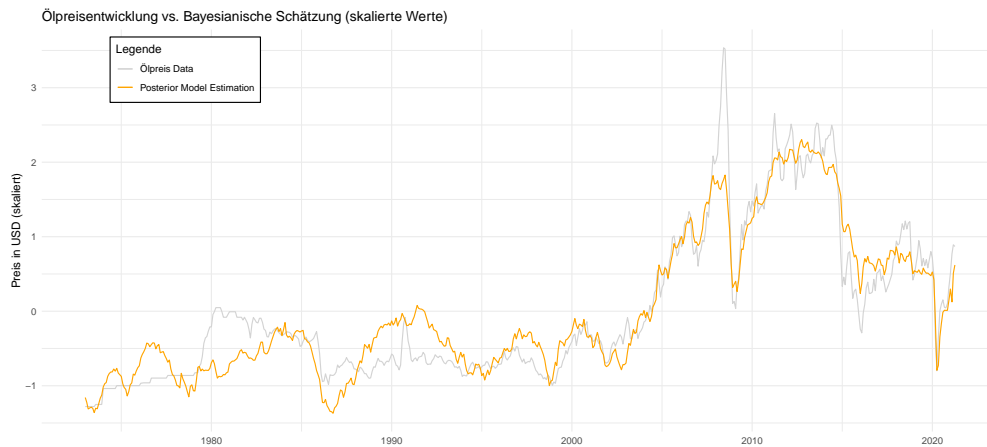


Figure 14: Daten vs. bayesianische Schätzung ($\tau_0 = 0.000001, \tau_1 = 10$)

Die Gegenüberstellung der Vorhersage mit den tatsächlichen Daten in Bild 14 zeigt, dass sich die Schätzung nun verschlechtert. Der MSE sowie AIC steigen.

Gütemaß	Wert des Informationskriteriums
MSE	0.1209092
AIC	2,477.181

Nachfolgende Statistiken betreffend Modell 2 bestätigen den Verdacht aus Bild 13.

```
## Coefficients:
##               Probability           Mean           SD
## CPIAUCSL          1 -2.146332986584  0.145623530
## CUSR0000SAC        1  2.475074517326  0.114847186
## S.P.500            0  0.000015695129  0.001019631
## IPFUELS            1  0.289947617928  0.051470766
## IPMAT              1  0.448079929559  0.066954282
## CMRMTSPLx          0  0.000003100992  0.001014523
## TWEXAFEGSMTHx      1  0.275382677830  0.023663946
## EXCAUSx            1 -0.652323379006  0.022341041
##
## N: 580  Burins: 1000  Draws: 2000
## Tau0:  0.000001  Tau1:  10
```

3.4.3 Hyperparameter: $\tau_0 = 50, \tau_1 = 10$

Wird τ_0 sehr hoch gesetzt, würde ich erwarten, dass die Koeffizienten nicht mehr vernünftig gegen 0 geshrinkt werden können, da die Präzision der Prior Varianz für den Fall $\beta_j = 0$ zu niedrig ist.

```
set.seed(2)
model3 = SSVS(X = FRED2[, 2:9], y= FRED2$OILPRICE, tau0 = 50, tau1 = 10, scale = TRUE)
```

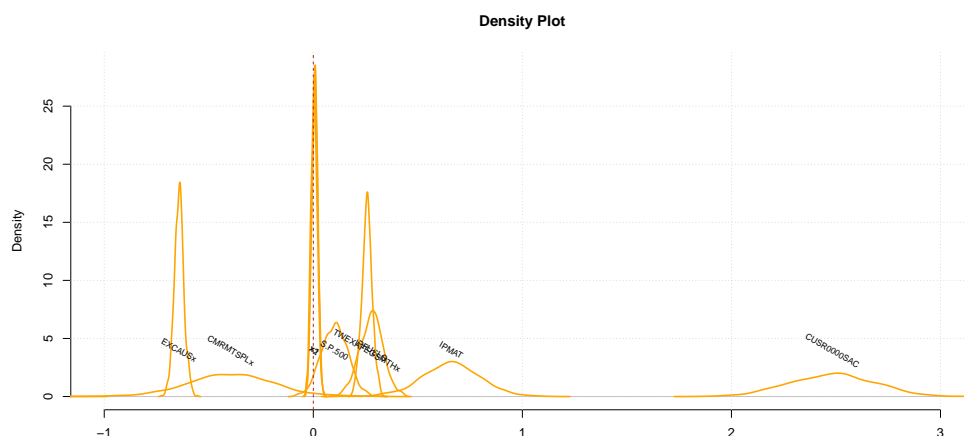


Figure 15: Density Plot $\tau_0 = 50, \tau_1 = 10$

Bild 15, sowie Bild 16 zeigen, dass nun alle Variablen im Modell verbleiben. Es fällt dem Algorithmus also sichtlich schwer Variablenselektion zu betreiben, da die Präzision wie eingangs erwähnt zu niedrig ist.

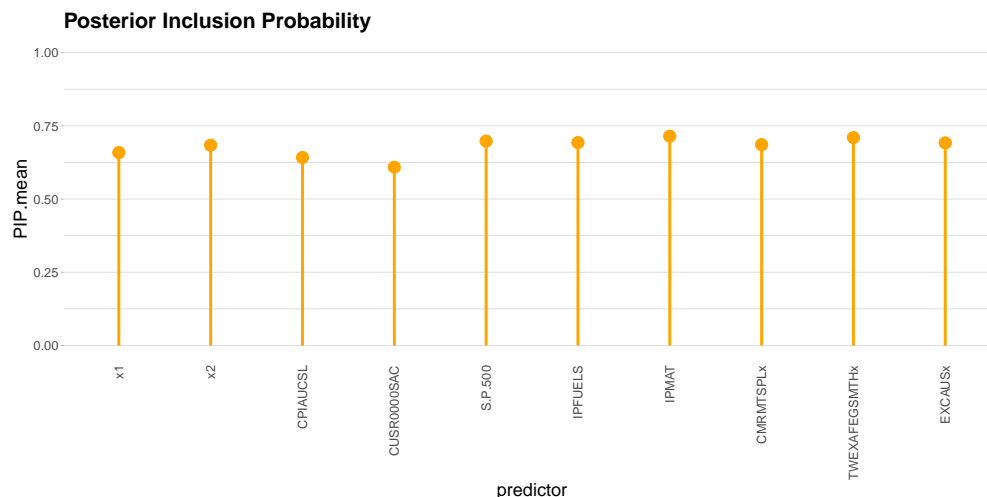


Figure 16: Posterior Inclusion Probability ($\tau_0 = 50, \tau_1 = 10$)

Hierbei würden alle Variablen in über 50% der Fälle im Modell verbleiben, da die Posterior Inclusion Probability für alle Koeffizienten größer als 0.5 ist. SSVS ist es mit dieser hohen Prior Varianz ($\tau_0 = 50$ für den Fall $\beta_j = 0$) nicht möglich Koeffizienten nahe 0 zu setzen. Somit findet offensichtlich keine effektive Variablenselektion statt, da wie eingangs befürchtet τ_0 zu hoch gesetzt wurde.

Bild 17 zeigt die Dichte der im Modell enthaltenen Koeffizientenschätzer:

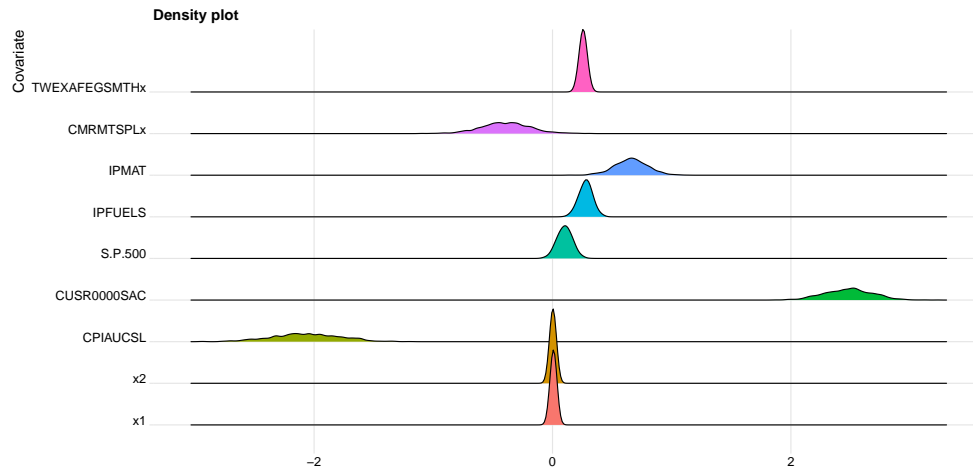


Figure 17: Dichte aller im Modell berücksichtigter Koeffizienten ($\tau_0 = 50, \tau_1 = 10$)

Bild 18 zeigt uns erneut die Schätzung basierend auf dem bayesianischen Modell (diesmal jedoch überraschenderweise mit einem, im Vergleich zu model1 und model2, recht ähnlichen MSE):

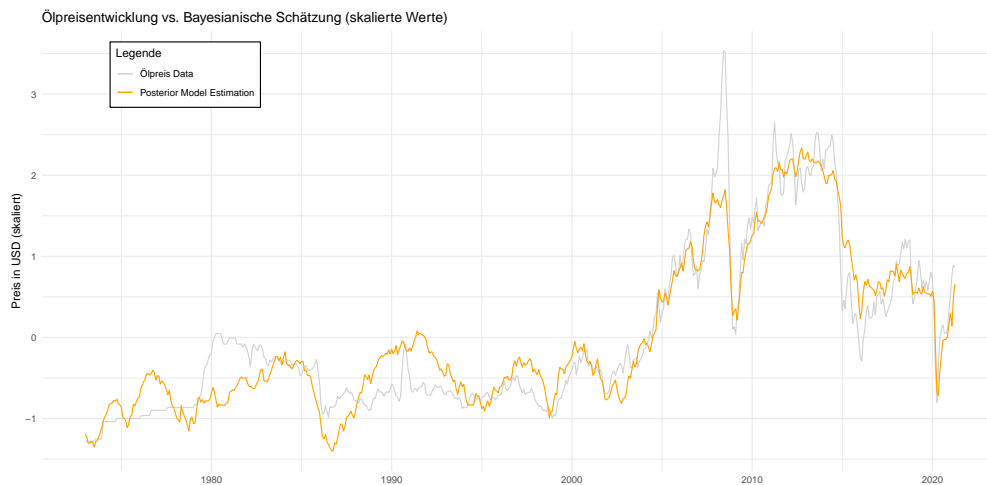


Figure 18: Daten vs. bayesianische Schätzung ($\tau_0 = 0.000001, \tau_1 = 10$)

Obwohl der MSE nun für dieses Modell sinkt, ist klar ersichtlich dass durch die gestiegene Modellkomplexität (höhere Parameteranzahl), aufgrund des absorbing states, das AIC steigt. Somit steigt die Modellkomplexität überproportional zu den RSS und verschlechtern das Modell insgesamt.

Gütemaß	Wert des Informationskriteriums
MSE	0.119851
AIC	2,480.083

Nachfolgende Statistik zeigt erneut die numerischen Werte betreffend Modell 3. Hierbei wird nochmal deutlich ersichtlich, dass SSVS es nicht schafft effektive Variablenselektion zu betreiben.

```
## Coefficients:
##              Probability      Mean      SD
## CPIAUCSL          0.672 -2.0707505 0.27352263
## CUSR0000SAC        0.641  2.4868342 0.19295623
## S.P.500            0.705  0.1017159 0.06148184
## IPFUELS            0.655  0.2790482 0.05456835
## IPMAT              0.694  0.6556497 0.14440011
## CMRMTSPLx          0.679 -0.3881098 0.21696524
## TWEXAFEGSMTHx      0.683  0.2583424 0.02544406
## EXCAUSx            0.689 -0.6409142 0.02382877
##
## N: 580  Burins: 1000  Draws: 2000
## Tau0:  50  Tau1:  10
```

3.4.4 Hyperparameter: $\tau_0 = 0.1, \tau_1 = 100$

Um ein Modell zu finden, welches den bisher geringsten MSE generiert, erfolgt nun ein Hyperparameter Testing mit verschiedenen Werten für τ_0, τ_1 sowie der Anzahl der Ziehungen S :

```
set.seed(2)
for (i in c(0.1, 0.01, 0.001, 0.0001)){
  for (j in c(10, 100, 1000, 1000)){
    for (z in c(1000, 5000)){
      testrun = SSVS(X = FRED2[, 2:9], y= FRED2$OILPRICE, tau0 = i, tau1 = j,
                    scale = TRUE, nsave = z, nburn = z)
      cat("tau0:      ", i, "\n")
      cat("tau1:      ", j, "\n")
      cat("nsave/nburn: ", z, "\n")
      print(testrun[[5]])
      print("-----")
    }
  }
}
```

```
## tau0:      0.1
## tau1:      10
## nsave/nburn: 1000
##           MSE
## 0.1199707
## [1] "-----"
```

```
## tau0:      0.1
## tau1:      10
## nsave/nburn: 5000
##           MSE
## 0.1200288
## [1] "-----"
```

```
## tau0:      0.1
```

```

## tau1:          100
## nsave/nburn:  1000
##      MSE
## 0.1196582
## [1] "-----"

## tau0:          0.1
## tau1:          100
## nsave/nburn:  5000
##      MSE
## 0.1190266
## [1] "-----"

## tau0:          0.1
## tau1:          1000
## nsave/nburn:  1000
##      MSE
## 0.1198325
## [1] "-----"

## tau0:          0.1
## tau1:          1000
## nsave/nburn:  5000
##      MSE
## 0.1199378
## [1] "-----"

## tau0:          0.1
## tau1:          1000
## nsave/nburn:  1000
##      MSE
## 0.119996
## [1] "-----"

## tau0:          0.1
## tau1:          1000
## nsave/nburn:  5000
##      MSE
## 0.1194704
## [1] "-----"

## tau0:          0.01
## tau1:          10
## nsave/nburn:  1000
##      MSE
## 0.1193799
## [1] "-----"

## tau0:          0.01
## tau1:          10
## nsave/nburn:  5000
##      MSE
## 0.120264
## [1] "-----"

```



```
## tau0:          0.01
## tau1:          100
## nsave/nburn:  1000
##      MSE
## 0.1203911
## [1] "-----"
```

```
## tau0:          0.01
## tau1:          100
## nsave/nburn:  5000
##      MSE
## 0.1201186
## [1] "-----"
```

```
## tau0:          0.01
## tau1:          1000
## nsave/nburn:  1000
##      MSE
## 0.1200697
## [1] "-----"
```

```
## tau0:          0.01
## tau1:          1000
## nsave/nburn:  5000
##      MSE
## 0.1204173
## [1] "-----"
```

```
## tau0:          0.01
## tau1:          1000
## nsave/nburn:  1000
##      MSE
## 0.12045
## [1] "-----"
```

```
## tau0:          0.01
## tau1:          1000
## nsave/nburn:  5000
##      MSE
## 0.1204334
## [1] "-----"
```

```
## tau0:          0.001
## tau1:          10
## nsave/nburn:  1000
##      MSE
## 0.1205955
## [1] "-----"
```

```
## tau0:          0.001
## tau1:          10
## nsave/nburn:  5000
```

```

##      MSE
## 0.120204
## [1] "-----"

## tau0:      0.001
## tau1:      100
## nsave/nburn: 1000
##      MSE
## 0.1205638
## [1] "-----"

## tau0:      0.001
## tau1:      100
## nsave/nburn: 5000
##      MSE
## 0.1200392
## [1] "-----"

## tau0:      0.001
## tau1:      1000
## nsave/nburn: 1000
##      MSE
## 0.1203482
## [1] "-----"

## tau0:      0.001
## tau1:      1000
## nsave/nburn: 5000
##      MSE
## 0.1206268
## [1] "-----"

## tau0:      0.001
## tau1:      1000
## nsave/nburn: 1000
##      MSE
## 0.1203778
## [1] "-----"

## tau0:      0.001
## tau1:      1000
## nsave/nburn: 5000
##      MSE
## 0.1199093
## [1] "-----"

## tau0:      0.0001
## tau1:      10
## nsave/nburn: 1000
##      MSE
## 0.1206306
## [1] "-----"

```

```
## tau0:          0.0001
## tau1:          10
## nsave/nburn:   5000
##      MSE
## 0.1207574
## [1] "-----"
```

```
## tau0:          0.0001
## tau1:          100
## nsave/nburn:   1000
##      MSE
## 0.1207201
## [1] "-----"
```

```
## tau0:          0.0001
## tau1:          100
## nsave/nburn:   5000
##      MSE
## 0.1208597
## [1] "-----"
```

```
## tau0:          0.0001
## tau1:          1000
## nsave/nburn:   1000
##      MSE
## 0.1207624
## [1] "-----"
```

```
## tau0:          0.0001
## tau1:          1000
## nsave/nburn:   5000
##      MSE
## 0.1208617
## [1] "-----"
```

```
## tau0:          0.0001
## tau1:          1000
## nsave/nburn:   1000
##      MSE
## 0.120653
## [1] "-----"
```

```
## tau0:          0.0001
## tau1:          1000
## nsave/nburn:   5000
##      MSE
## 0.1200052
## [1] "-----"
```

Wir sehen, dass wir den geringsten MSE für ein Modell bekommen, für das die Prior Varianz τ_0 - für den Fall $\beta_j = 0 - 0.1$ und τ_1 - für den Fall $\beta_j \neq 0 - 100$ beträgt. Weiterhin erhöhen wir den Burn-In S_0 sowie die abgespeicherten Ziehungen S_1 von 1,000 auf 5,000. Jenes Modell wird nun geschätzt:

```
set.seed(2)
model4 = SSVS(X = FRED2[, 2:9], y = FRED2$OILPRICE, tau0 = 0.1, tau1 = 100,
              nburn = 5000, nsave = 5000, scale = TRUE)
```

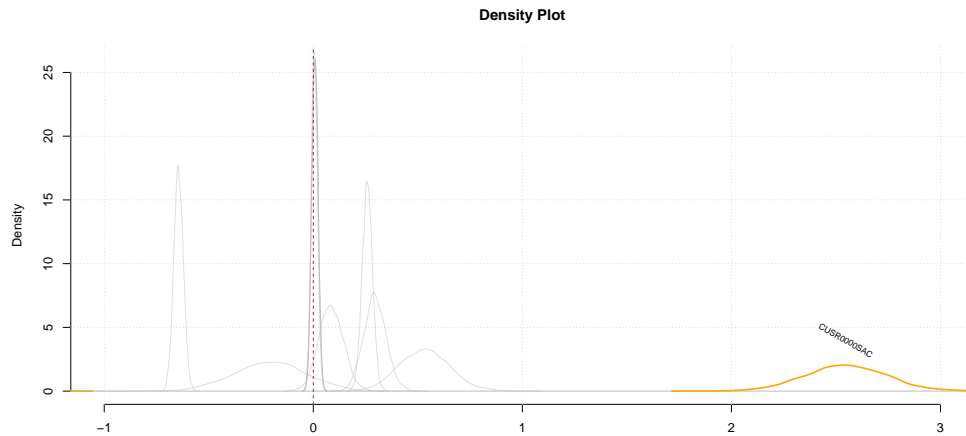


Figure 19: Density Plot $\tau_0 = 0.001, \tau_1 = 1000$

Unter Verwendung dieser Prior Varianzen sind nurnoch die Variablen CPIAUCSL und CUSR0000SAC zu 100% in den Modellen enthalten. Obwohl τ_0 meiner Meinung nach mit 0.1 doch recht hoch angesetzt ist, erhalten wir laut Bild 20 eine deutlich sichtbare Variablenselektion. So ist z.B. die Variable EXCAUSx nur in nahe 25% der Fälle inkludiert.

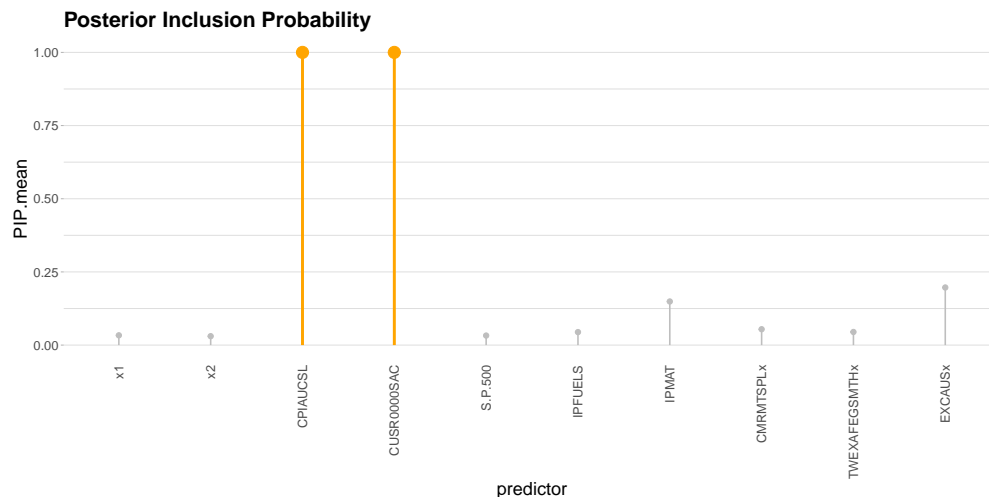


Figure 20: Posterior Inclusion Probability ($\tau_0 = 0.1, \tau_1 = 100$)

Wie zu Beginn dieses Unterkapitels gesehen, erhalten wir in diesem Fall den geringsten MSE für alle Vergleichsmodelle.

Da Bild 19, aufgrund der fixen Achsen, kein vollständiges Bild der Dichtefunktionen wiedergibt, sind Bild 21 alle Dichtefunktionen der Schätzkoeffizienten zu entnehmen.

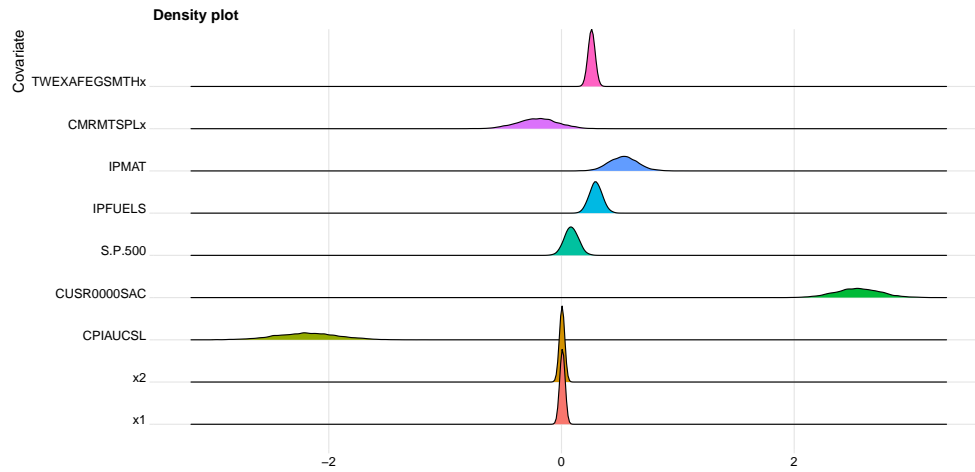


Figure 21: Dichte aller im Modell berücksichtigter Koeffizienten ($\tau_0 = 0.1, \tau_1 = 100$)

Unter Verwendung dieser Hyperparameter, erhalten wir insbesondere in den Vorhersagen ab den 1990er Jahren einen relativ guten fit (Bild 22).

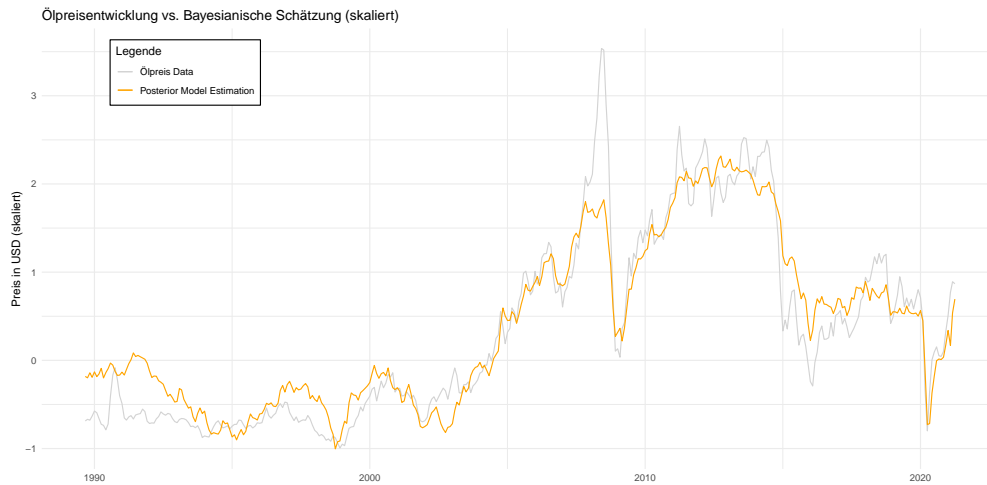


Figure 22: Fit ab den 1990er Jahren

Obwohl wir im Modell 4 einen leicht erhöhten MSE erhalten, erhalten wir im Modell den bisher geringsten AIC. Somit würde dieses Modell, das bisher am meisten präferierte Modell darstellen.

Gütemaß	Wert des Informationskriteriums
MSE	0.1200228
AIC	2,466.111

Abschließend die Statistiken zu Modell 4:

```
## Coefficients:
##              Probability          Mean          SD
## CPIAUCSL      1.0000 -2.17640076  0.26688977
## CUSR0000SAC    1.0000  2.54096792  0.19356418
## S.P.500        0.0322  0.08414584  0.05922712
## IPFUELS        0.0480  0.29077294  0.05448458
## IPMAT          0.1426  0.53493509  0.11920501
## CMRMTSPLx      0.0472 -0.20938308  0.17088767
## TWEXAFEGSMTHx  0.0424  0.25954016  0.02476850
## EXCAUSx        0.1942 -0.64422671  0.02309200
##
## N: 580  Burins: 5000  Draws: 10000
## Tau0:  0.1  Tau1:  100
```

3.4.5 Special Topic: Modell ab 2000er Jahre

Da wir bei der Gegenüberstellung der Vorhersagen (fitted Values) mit den tatsächlichen Werten erkennen konnten, dass die Vorhersagen insbesondere ab den 2000er Jahren einen deutlich besseren Fit hatten, könnte man vermuten, dass die erklärenden Variablen erst ab diesem Zeitraum deutlich zum Erklärungsgehalt des Ölpreises beitragen.

Um diese Vermutung empirisch zu determinieren, wird im Nachfolgenden versucht, das Modell erneut auf ein Subset der Daten (ca. ab 2000er Jahre) zu rechnen. Hierfür werden die Daten zunächst eingeschränkt und anschließend auf das neue Datenset, mit Hilfe der gleichen Hyperparameter wie in Kapitel 3.4.4, gerechnet:

```
# Datensatz einschränken
FRED3 = FRED2[300:580, ]

# Modell 5
set.seed(2)
model5 = SSVS(X = FRED3[, 2:9], y = FRED3$OILPRICE, tau0 = 0.1, tau1 = 100,
              nburn = 5000, nsave = 5000, scale = TRUE)
```

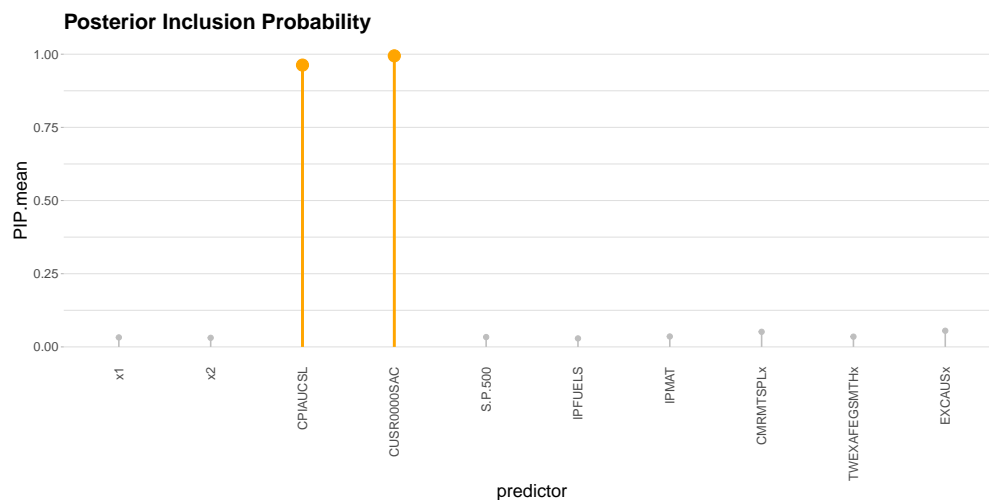


Figure 23: Posterior Inclusion Probability ($\tau_0 = 0.1, \tau_1 = 100$) für Observationen ab 2000

Bei den Posterior Inclusion Probabilities zeigen sich leichte Unterschiede zu dem Modell aus Kapitel 3.4.4. Bild 23 zeigt, dass nach wie vor die Variablen CPIAUCSL (CPI: All Items) und CUSR0000SAC (CPI: Commodities) die höchsten PIBs aufweisen.

In Bild 24 sehen wir das Ergebnis der Eingangs aufgestellten Behauptung. Werden ausschließlich Daten ab den 2000er Jahren zum Fit des Modells verwendet, erhalten wir einen deutlich passenderen Fit.

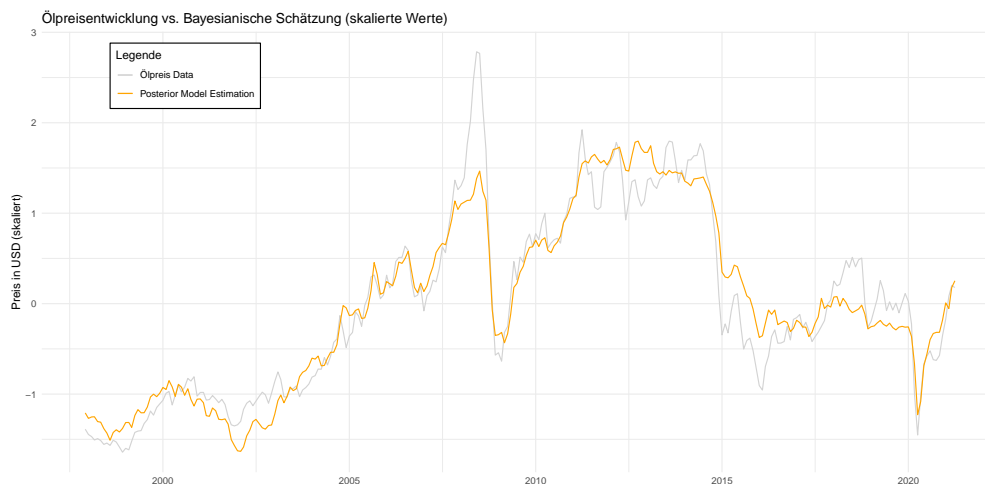


Figure 24: Model fitting für die Jahre ab 2000

Die Informationskriterien untermauern die Erkenntnisse aus Bild 24 zusätzlich.

Gütemaß	Wert des Informationskriteriums
MSE	0.08972493
AIC	993.2169

4 Conclusion

Kapitel 3.4 zeigt, dass die Auswahl der Hyperparameter ein entscheidender Faktor für die Güte des Modells ist. Hierbei gilt es dringend abzuwägen, in welcher Höhe die unterschiedlichen Parameter gesetzt werden sollen.

Setzt man beispielsweise Werte für τ_0 zu klein, wandert der Gibbs Sampler in den absorbing state, wird der Wert für τ_0 zu hoch gesetzt, gibt es Schwierigkeiten beim Shrinking der Koeffizienten. Tau muss daher sorgfältig abgestimmt werden, um SSVS effizient umzusetzen und gleichzeitig einen guten Proxy für die ‘Posterior Model Probabilities’ der Variablenselektion, auf Grundlage der ‘point-zero null hypotheses’, zu haben. George und McCulloch beschreiben in ihrem Paper einen ‘semiautomatic approach’ um geeignete Werte für Tau zu finden. Dieser übersteigt jedoch den Rahmen der Projektarbeit und wurde daher nicht weiter thematisiert.

Alles in allem stellt die richtige Auswahl der Hyperparameter wohl eine Kunst für sich innerhalb der bayesianischen Statistik dar. Sind sie jedoch richtig gewählt, erhalten wir mit einem recht einfach gestrickten Modell zuverlässige Ergebnisse!

5 Reference

- [1] Casella, G. and George, E.I. (1992) Explaining the Gibbs Sampler. The American Statistician
- [2] George, E.I. and McCulloch, R.E. (1993) Variable selection via Gibbs sampling. J. Am. Stat. Assoc.
- [3] Huber, F. (2021) Lecutre Notes (Sommersemester 2021)
- [4] Koop, G. (2003) Bayesian Econometrics