

Stochastic Modelling

Zusammenfassung v1.1

**Kälin Thomas, MSE ICT
HS 08/09**

1. PROBABILITY REVIEW	4
1.1. Conditional Probabilities	4
1.2. The Law of total Probabilities	4
1.3. Independent Events	4
1.4. Random Variables	4
1.5. Discrete Random Variables	5
1.6. Continuous Random Variables	6
1.7. Normal RV	6
1.8. Jointly Distributed Random Variables	7
1.8.1. Covariance	7
1.8.2. Not Correlated und Independent	7
1.9. Limit Theorems	8
1.9.1. The Law of large Numbers	8
1.9.2. Central Limit Theorem	8
1.10. Conditional Probabilities – Teil 2	8
2. STOCHASTISCHE PROZESSE	9
2.1. Grundlagen	9
2.1.1. Klassifizierung von stochastischen Prozessen	9
2.1.2. Kennzahlen von stochastischen Prozessen	9
2.1.3. Stationarität	10
2.1.4. Ergodizität	10
2.2. Power Spectral Density	11
2.3. Weisses Rauschen	11
2.3.1. Gaussches weisses Rauschen	12
2.4. Farbiges Rauschen	12
2.4.1. Der zufällige Telegraph Prozess	13
2.4.2. Beispiel: Produktionsmaschine	13
3. MARKOV KETTEN	14
3.1. Einleitung	14
3.2. Zeitdiskrete Markov-Ketten	14
3.2.1. Grundlagen	14
3.2.2. Chapman-Kolmogorov Gleichung	14
3.2.3. Stationäre Verteilung	15
3.2.4. Erreichbare und kommunizierende Zustände, irreduzible Prozesse	15
3.2.5. Klassen	16
3.2.6. Rekurrente und transiente Zustände	16
3.2.7. Anzahl stationärer Verteilungen bestimmen	16
3.3. Random Walk auf Graphen	16
3.3.1. Anwendungsbeispiel für Graphen: Bild-Segmentierung	17
3.4. Hidden Markov Model	18
3.4.1. Forward-Backward Algorithmus	18
3.4.2. Viterbi-Algorithmus	20
3.4.3. Baum-Welch-Algorithmus	20
3.4.4. Beispiel: Produktionsmaschine	21
3.5. Hidden Markov Models – Spracherkennung	23
3.5.1. Schritt 1 – Erstellen des Codebooks	23
3.5.2. Schritt 2 – Trainieren der Modelle	24
3.5.3. Schritt 3 – Erkennen der Wörter	24
3.5.4. Verbesserungsmöglichkeiten	25
3.5.5. Aktuelle Systeme	25
3.6. Zeitstetige Markovketten	25
3.6.1. Haltezeit	25
3.6.2. Memoryless Property	25
3.6.3. Erlang Verteilung	26
3.6.4. Eigenschaften einer zeitstetigen Markovkette	26
3.6.5. Chapman-Kolmogorov	26
3.7. Zählprozess	27

3.8. Poisson Prozess	27
3.8.1. Definition 1 – Betrachtung der Poissonverteilung	27
3.8.2. Definition 2 – Betrachtung des Verhaltens für kleine Zeitabschnitte	28
3.8.3. Definition 3 – Betrachtung der exponentialverteilten Haltezeiten	28
3.8.4. Zusammenführen und Splitten	28
3.8.5. Beispiele zum Poissonprozess	29
3.9. Geburts- und Sterbeprozesse	30

1. PROBABILITY REVIEW

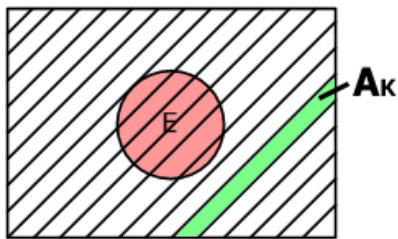
1.1. Conditional Probabilities

Die Wahrscheinlichkeit, dass E eintritt, unter der Voraussetzung, dass F eingetreten ist. Der Term unter dem Bruchstrich verkleinert den Ereignisraum.

$$P(E|F) = \frac{P(E \cap F)}{P(F)} = \frac{P(E, F)}{P(F)} \quad P(E \cap F) = P(E|F) * P(F) = P(F|E) * P(E)$$

1.2. The Law of total Probabilities

Wird auch „Konditionieren“ oder „Salami-Taktik“ genannt. Dabei wird der Ereignisraum in kleine „Scheiben“ aufgeteilt.



$$P(E) = \sum_{k=1}^n P(E|A_k)P(A_k)$$

1.3. Independent Events

E und F sind unabhängig, wenn das Wissen, dass F aufgetreten ist, die Wahrscheinlichkeit von E nicht beeinflusst.

$$P(E \cap F) = P(E|F) * P(F) = P(E) * P(F)$$

1.4. Random Variables

Eine Zufallsvariable X ist eine Funktion, welche den Ergebnissen des Ergebnisraums Werte aus den realen Zahlen zuordnet.

$$X : \Omega \rightarrow R$$

Discrete Random Variables	Continuous Random Variables
<p>Abbildung 1: Probability Mass Function</p> $P(X = x_i) = Wkeit$ $\sum_{i=1}^{\infty} P(X = x_i) = 1$	<p>Abbildung 2: Probability Density Function</p> $P(X \in [a; b]) = \int_a^b f(x) dx$ $\int_{-\infty}^{\infty} f(x) dx = 1$
Cumulative Distribution Function	
$F(X) = P(X \leq x) = \sum_{i=1}^x P(X = x_i)$	$F(X) = P(X \leq x) = \int_{-\infty}^x f(x) dx$

Expectation E(X)	
$E(X) = \sum_{i=1}^{\infty} x_i P(X = x_i)$	$E(X) = \int_{-\infty}^{\infty} xf(x)dx$
Variance V(X)	
$Var(X) = E(X^2) - E(X)^2$	$Var(X) = E(X^2) - E(X)^2$
Moment Generating Function $\phi(t)$	
$\phi(t) = E(e^{tX}) = \sum_{i=1}^{\infty} e^{tx_i} P(X = x_i)$	$\phi(t) = E(e^{tX}) = \int_{-\infty}^{\infty} e^{tx} f(x)dx$
<p>Durch Ableitung und Auswertung an $t=0$ kann aus der Moment Generating Function der Erwartungswert $E(X)$ bestimmt werden.</p> $E(X) = \left. \frac{d\phi(t)}{dt} \right _{t=0} \quad E(X^2) = \left. \frac{d^2\phi(t)}{dt^2} \right _{t=0}$	

1.5. Discrete Random Variables

Bernoulli RV (X(Success) = 1, X(Failure) = 0)			
$P(X)$	$P(X = 1) = p = 1 - P(X = 0)$	$E(X)$	$E(X) = p$
$VAR(X)$	$VAR(X) = p(1 - p)$	$\phi(t)$	$\phi(t) = e^t p + 1 - p$

Binomial RV (X ~ Binom(n,p))			
$P(X)$	$P(X = i) = \binom{n}{i} p^i (1 - p)^{n-i}$	$E(X)$	$E(X) = np$
$VAR(X)$	$VAR(X) = np(1 - p)$	$\phi(t)$	$\phi(t) = (pe^t + 1 - p)^n$

Poisson RV (X ~ Pois(λ))			
$P(X)$	$P(X = i) = \frac{\lambda^i}{i!} e^{-\lambda}$	$E(X)$	$E(X) = \lambda$
$VAR(X)$	$VAR(X) = \lambda$	$\phi(t)$	$\phi(t) = e^{\lambda(e^t - 1)}$

Geometric RV (X ~ Geom(p))			
$P(X)$	$P(X = i) = (1 - p)^{i-1} p$	$E(X)$	$E(X) = \frac{1}{p}$
$VAR(X)$	$VAR(X) = \frac{1 - p}{p^2}$	$\phi(t)$	$\phi(t) = \frac{pe^t}{1 - (1 - p)e^t}$

1.6. Continuous Random Variables

Uniform RV ($X \sim U(\alpha, \beta)$)			
$P(X)$	$f(x) = \frac{1}{\beta - \alpha}$	$E(X)$	$E(X) = \frac{\alpha + \beta}{2}$
$VAR(X)$	$VAR(X) = \frac{(\alpha - \beta)^2}{12}$	$\phi(t)$	$\phi(t) = \frac{e^{t\beta} - e^{t\alpha}}{t(\beta - \alpha)}$

Exponential RV ($X \sim \text{Exp}(\lambda)$)			
$P(X)$	$f(x) = \lambda e^{-\lambda x}$	$E(X)$	$E(X) = \frac{1}{\lambda}$
$VAR(X)$	$VAR(X) = \frac{1}{\lambda^2}$	$\phi(t)$	$\phi(t) = \frac{\lambda}{\lambda - t}$

1.7. Normal RV

Normal RV ($X \sim N(\mu, \sigma^2)$)			
$P(X)$	$f(x) = \frac{1}{\sqrt{2\pi\sigma}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right)$	$E(X)$	$E(X) = \mu$
$VAR(X)$	$VAR(X) = \sigma^2$	$\phi(t)$	$\phi(t) = \exp\left(\frac{\sigma^2 t^2}{2} + \mu t\right)$

Die normalisierte Standardnormalverteilung besitzt die Werte $\mu = 0$ und $\sigma = 1$. Da Das Integral nicht elementar gelöst werden kann gibt es passende Tabellen für die Umrechnung.

$$Y = \frac{X - \mu}{\sigma} \sim N(0,1)$$

$$\Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-y^2/2} dy$$

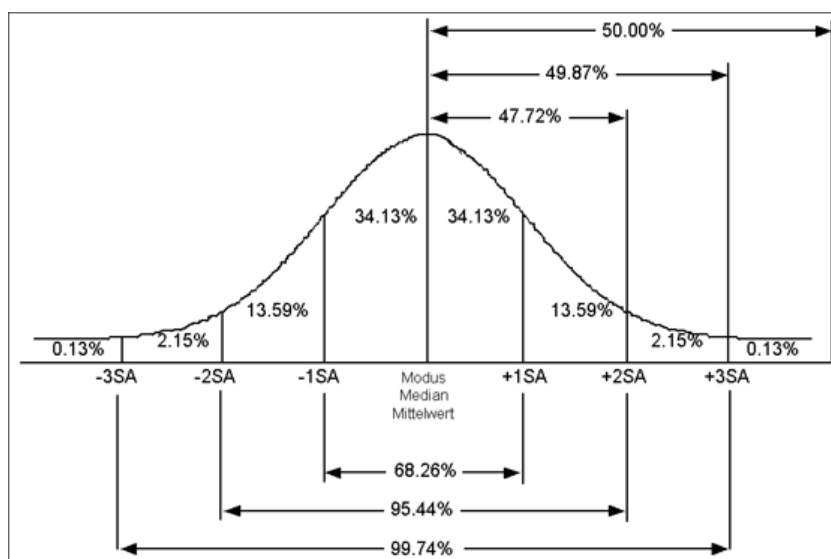


Abbildung 3: Standardnormalverteilung

1.8. Jointly Distributed Random Variables

Bis anhin wurden immer nur einzelne Zufallsvariablen betrachtet. Diese können aber auch kombiniert werden. Die „Joint Probability Distribution“ enthält anschliessend mehr Informationen, als eine einzelne Verteilung alleine.

$$F(a,b) = P(X \leq a, Y \leq b)$$

Joint Probability Mass Function (Discrete)	Joint Probability Density Function (Continuous)
$p(x, y) = P(X = x, Y = y)$	$p(X \in [a,b], Y \in [c,d]) = \int_a^b \int_c^d f(x, y) dy dx$

1.8.1. Covariance

Die Kovarianz ist in der Statistik eine nichtstandardisierte Masszahl für den Zusammenhang zweier statistischer Merkmale. Die Kovarianz gibt zwar die Richtung einer Beziehung zwischen zwei Variablen an, über die Stärke des Zusammenhangs wird aber keine Aussage getroffen. Die Kovarianz ist eine Verallgemeinerung der Varianz, denn es gilt: $Cov(X, X) = VAR(X)$.

Allgemein	$Cov(X, Y) = E((X - E(X)) * (Y - E(Y))) = E(XY) - E(X)E(Y)$
Discrete	$Cov(X, Y) = \sum_{xi, xj} (xi - E(X)) * (yj - E(Y)) * P(X = xi, Y = yj)$
Continuous	$Cov(X, Y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (x - E(X)) * (y - E(Y)) * f(x, y) dy dx$

Die Kovarianz zweier unabhängiger, beziehungsweise unkorrelierter Zufallsvariablen ist 0. Sind zwei Zufallsvariablen positiv korreliert, so tendiert die Variable X gleichmässig zu Y zu steigen. Sind zwei Zufallsvariablen negativ korreliert, so tendiert die Variable X gleichmässig zu Y zu sinken.

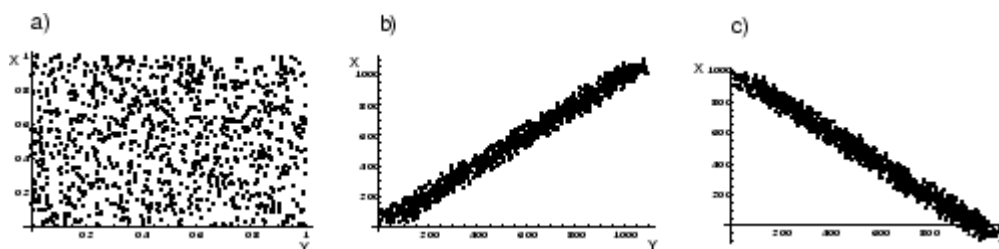


Abbildung 4: Unkorreliert, positiv korreliert, negativ korreliert

1.8.2. Not Correlated und Independent

Not Correlated	$E(XY) = E(X) * E(Y)$
Independent	$P(X = a, Y = b) = P(X = a) * P(Y = b)$ $f(a, b) = f_x(a) * f_y(b)$

1.9. Limit Theorems

1.9.1. The Law of large Numbers

Der Durchschnitt einer Reihe unabhängiger Zufallsvariablen mit denselben Verteilungen konvergiert mit der Wahrscheinlichkeit 1 zum Mittelwert (=Erwartungswert) dieser Verteilung.

$$\frac{X_1 + X_2 + \dots + X_n}{n} \rightarrow_{\infty} \mu = E(X)$$

1.9.2. Central Limit Theorem

Das zentrale Grenzwert Theorem sagt aus, dass eine grosse Menge an unabhängiger und ähnlich verteilter RV ungefähr der Normalverteilung folgt.

$$\frac{X_1 + X_2 + \dots + X_n - n\mu}{\sigma\sqrt{n}}$$

$$z = \frac{\bar{x} - \mu}{\sigma} \sim N(0,1) \quad \bar{x} = \frac{\sum X_i}{n} \quad \mu = \frac{\sum \mu_i}{n} \quad \sigma = \frac{\sigma_i}{\sqrt{n}}$$

1.10. Conditional Probabilities – Teil 2

Die bedingten Wahrscheinlichkeiten für zwei Ereignisse können auf Zufallsvariablen erweitert werden. Es gelten neu folgende Regeln:

Conditional Probability Mass Function	$P_{X Y}(x y) = P(X = x Y = y) = \frac{P(X = x, Y = y)}{P(Y = y)}$
Conditional Cumulative Distribution Function	$F_{X Y}(x y) = P(X \leq x Y = y) = \sum_{a \leq x} \frac{P(X = a, Y = y)}{P(Y = y)}$
Conditional Expectation	$E(X Y = y) = \sum_x x P(X = x Y = y) = \sum_x x \frac{P(X = x, Y = y)}{P(Y = y)}$

Conditional Density Function	$f_{X Y}(x y) = P(X = x Y = y) = \frac{f(x, y)}{f_Y(y)}$
Conditional Cumulative Distribution Function	$F_{X Y}(x y) = P(X \leq x Y = y) = \int_{-\infty}^x \frac{f(x', y)}{f_Y(y)} dx'$
Conditional Expectation	$E(X Y = y) = \int_{-\infty}^{\infty} x f_{X Y}(x y) dx = \int_{-\infty}^{\infty} x \frac{f(x, y)}{f_Y(y)} dx$

2. STOCHASTISCHE PROZESSE

2.1. Grundlagen

Ein stochastischer Prozess X ordnet einem Index t Werte aus einem Zustandsraum S zu. In den meisten Anwendungen ist T eine Teilmenge von R und repräsentiert die Zeit. X_t entspricht der Position im Zustandsraum S zum Zeitpunkt t , X_0 ist der Startpunkt.

$$X_t = \Omega \rightarrow S \quad t \in T$$

2.1.1. Klassifizierung von stochastischen Prozessen

	T diskret	T stetig
S diskret	Random Walk	Poisson Process
S stetig	Time Series	Brownian Motion

2.1.2. Kennzahlen von stochastischen Prozessen

Wie der Erwartungswert oder die Varianz einer normalen Zufallsvariable einen guten Eindruck über die Verteilung vermitteln, so können auch für stochastische Prozesse einige wichtige Kennwerte zur besseren Erfassung des Prozesses berechnet werden:

Mean	$E(X_t)$ für jedes $t \in T$
Autocorrelation Function	$R_x(t_1, t_2) = E[X_{t_1} X_{t_2}] = E(X_{t_1}) * E(X_{t_2})$
Autocovariance Function	$C_x(t_1, t_2) = R_x(t_1, t_2) - E[X_{t_1}] * E[X_{t_2}]$
Correlation Coefficient	$\rho_x(t_1, t_2) = \frac{C_x(t_1, t_2)}{\sqrt{C_x(t_1, t_1) * C_x(t_2, t_2)}}$
Average Power	$R_x(t, t) = E[X_t^2]$

Die **AUTOKORRELATION** ist ein Begriff aus der Statistik und der Signalverarbeitung. Im statistischen Modell geht man von einer geordneten Folge von Zufallsvariablen aus. Vergleicht man die Folge mit sich selbst, so spricht man von Autokorrelation. Da jede unverschobene Folge mit sich selbst am ähnlichsten ist, hat die Autokorrelation für die unverschobenen Folgen den höchsten Wert. Wenn zwischen den Gliedern der Folge eine Beziehung besteht, die mehr als zufällig ist, hat auch die Korrelation der ursprünglichen Folge mit der verschobenen Folge in der Regel einen Wert, der signifikant von Null abweicht. Man sagt dann, die Glieder der Folge sind autokorreliert. Mit Hilfe der Autokorrelation ist es möglich, Zusammenhänge zwischen den beobachteten Ergebnissen zu verschiedenen Beobachtungszeitpunkten einer Messreihe festzustellen.

Die **AUTOKOVARIANZFUNKTION** untersucht im Gegensatz zur Kovarianz (Kapitel 1.8.1) nicht den Zusammenhang zwischen zwei verschiedenen Zufallsvariablen, sondern zwischen Realisationen der gleichen Zufallsvariablen zu unterschiedlichen Zeitpunkten. Es handelt sich hierbei folglich um ein Mass des Zusammenhangs von Beobachtungsdaten und nicht der kompletten Zufallsvariablen.

Der **KORRELATIONSKOEFFIZIENT** ist ein dimensionsloses Mass für den Grad des linearen Zusammenhangs zwischen zwei Merkmalen. Er kann Werte zwischen -1 und 1 annehmen. Bei einem Wert von $+1$ (bzw. -1) besteht ein vollständig positiver (bzw. negativer) linearer Zusammenhang zwischen den betrachteten Merkmalen. Wenn der Korrelationskoeffizient den Wert 0 aufweist, hängen die beiden Merkmale überhaupt nicht linear voneinander ab. Allerdings können diese ungeachtet dessen in nicht-linearer Weise voneinander abhängen.

2.1.3. Stationarität

Stationarität ist eine der bedeutendsten Eigenschaften stochastischer Prozesse. Mit der Stationarität erhält man Eigenschaften, die nicht nur für einzelne Zeitpunkte gelten, sondern Invarianzen über die Zeit hinweg sind.

Independent Increments	Die Differenzen zwischen allen $(X_t - X_{t-1})$ sind für aufsteigende t unabhängig.
Stationary Increments	Die Verteilungen zwischen allen $(X_t - X_{t-1})$ -Paaren sind für alle Zeiten t identisch.
Strinct Sense Stationary [Im strengen Sinne stationär]	Die Verteilung ändert sich nicht für alle Zeiten t . Diese Bedingung ist technisch schwer überprüfbar. $X_{t=1} \dots X_{t=n} \stackrel{v}{=} X_{t=1+s} \dots X_{t=n+s}$
Wide Sense Stationary (WSS) [Im weiten Sinne stationär]	Der Mittelwert $E(X_t)$ ist konstant für alle t . Dazu hängt die Autokorrelationsfunktion $R_x(t_1, t_2)$ nur von der Differenz $t_2 - t_1$, nicht aber von der absoluten Zeit t ab.

Ist ein stochastischer Prozess WSS, so wird für die Autokorrelations-Funktion auch $R_x(t)$ anstelle von $R_x(t_1, t_2)$ geschrieben.

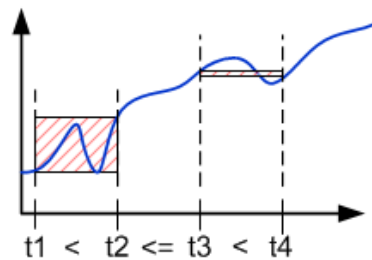


Abbildung 5: Beispiel von zwei unabhängigen Inkrementen

2.1.4. Ergodizität

In der Praxis ist die Autokorrelations-Funktion R_x nicht bekannt sondern muss anhand der verfügbaren Daten geschätzt werden. In der Statistik wird dabei zu einem bestimmten Zeitpunkt eine Stichprobe gezogen und deren Mittelwert berechnet. Bei einem stochastischen Prozess ist das aber nicht so einfach, da theoretisch aus mehreren Trajektorien ein Mittelwert bestimmt werden müsste – meistens ist aber nur eine Trajektorie vorhanden.

Aus diesem Grund ist es wichtig, dass der wahre Mittelwert unabhängig vom gewählten t ist (der Mean also immer gleich bleibt). Damit können wir nämlich eine Schätzung für den Mittelwert aus den Zuständen zu unterschiedlichen Zeitpunkten t einer einzigen Trajektorie bestimmen.

Ein stochastischer Prozess X_t mit zeit-invariantem Mittelwert $E(X_t)$ wird MEAN ERGODIC genannt.
Ein WSS stochastischer Prozess X_t wird MEAN ERGODIC genannt, wenn gilt: $C_X(0) < \infty \quad \text{und} \quad \lim_{t \rightarrow \infty} C_X(t) = 0$

BEISPIEL WÜRFEL (WIKIPEDIA): Das Mittel beim Würfeln kann sowohl dadurch ermittelt werden, dass man mit einem Würfel etwa 1000x hintereinander würfelt, als auch dadurch, dass man mit 1000 Würfeln gleichzeitig würfelt. Das Würfeln verhält sich also ergodisch.

2.2. Power Spectral Density

Dieses Kapitel benötigt ziemlich viele (mir fehlende) mathematische Kenntnisse. Aus diesem Grund müssen für die Prüfung nur Konzepte und keine genauen Berechnungen bekannt sein.

Die **SPEKTRALE LEISTUNGSDICHTE** eines WSS Prozess X_t ist die Fourier Transformation $S_X(k)$ seiner zugehörigen Autokorrelations-Funktion $R_X(t)$:

$$S_X(k) = \int_{-\infty}^{\infty} e^{-jkt} R_X(t) dt$$

Mit Hilfe der Power Spectral Density wird der Frequenzinhalt eines stochastischen Prozesses (oder eines Signals) erfasst. Das hilft beim Auffinden von Periodizitäten.

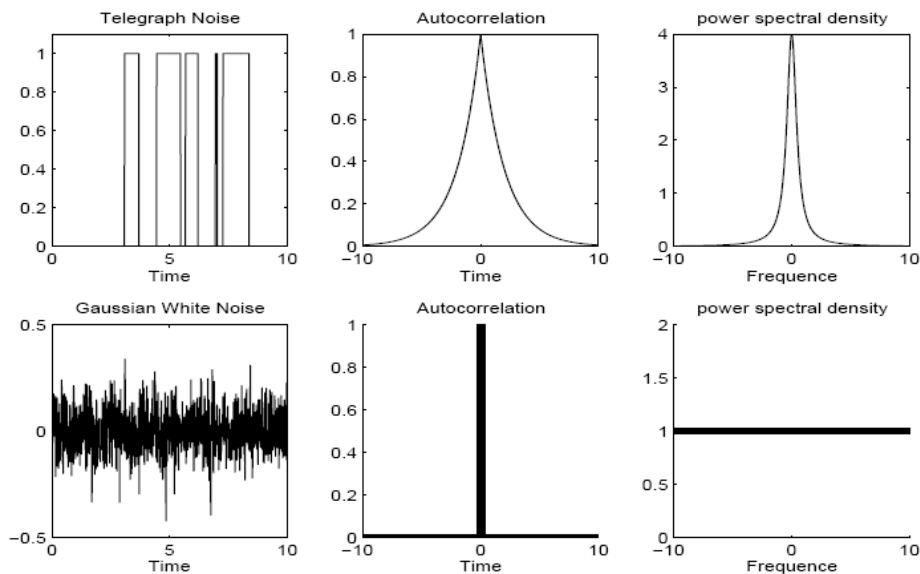


Abbildung 6: Autokorrelation und Power Spectral Density

Obige Grafik zeigt die Autokorrelation und Power Spectral Density für zwei Signale (bzw. stochastische Prozesse). Bei Signalen ohne Zusammenhang (unkorreliert) nimmt die Korrelation sehr schnell ab, beim weissen Rauschen geht sie sogar gleich wieder gegen 0, sofern nicht mit dem eigenen Wert korreliert wird. Wir sehen oben ausserdem, dass im weissen Rauschen jede Frequenz mit derselben Energie vorkommt.

2.3. Weisses Rauschen

Ein WSS stochastischer Prozess X_t wird weisses Rauschen genannt, wenn sein Erwartungswert $E(X_t) = 0$ und die Amplitude im Power Spectral Density $S_X(k)$ für alle Frequenzen konstant ist. Ein Beispiel dazu findet sich in Abbildung 6. Eine alternative, aber identische Definition lautet wie folgt:

Ein WSS stochastischer Prozess X_t wird „weisses Rauschen“ genannt, wenn gilt:

$$E(X_t) = 0 \quad \text{und} \quad R_X(t) = C_X(t) = \frac{D}{2} \delta(t)$$

Das weisse Rauschen stellt den einfachsten stochastischen Prozess dar, jedoch werden viele komplexere Prozesse und Zeitreihen aus solchen konstruiert.

Weisses Rauschen besitzt eine konstante Energie für alle Frequenzen, das Spektrum ist deshalb flach. Das ist analog zum weissen Licht, welches ebenfalls einen gleich bleibenden Anteil aller Farben eines Regenbogens besitzt. Wegen diesem flachen (und unendlichen!) Spektrum besitzt das weisse

Rauschen eine unendliche Energie und ist somit in der Realität nicht realisierbar. In der Praxis fällt beim weissen Rauschen die Leistungsdichte für sehr große Frequenzen ab. Dennoch ist es ein gutes Konzept als Annäherung an ähnliche Prozesse.

Die Autokorrelationsfunktion eines weissen Rauschens ist ein Vielfaches eines Dirac-Impuls bei 0. Diese Konstante wird im Kurs D/2 genannt. Das heisst, für beliebig kleine $\Delta t > 0$ gilt: Was zum Zeitpunkt $t + \Delta t$ geschieht, ist absolut unabhängig davon, was zum Zeitpunkt t geschehen ist (vollständige Unkorreliertheit). Auch hier sieht man, dass das weisse Rauschen so in seiner Absolutheit in der Realität nicht existieren kann.

2.3.1. Gaussches weisses Rauschen

Das gaussche weisse Rauschen besitzt die Dichtefunktion einer Normalverteilung. Das bedeutet, dass die Werte des Rauschens zu jedem Zeitpunkt normalverteilt sind. Grafisch kann man sich das so vorstellen, dass für jeden Zeitpunkt die Entscheidung für die „Fortführungsrichtung des Signals“ normalverteilt ist.

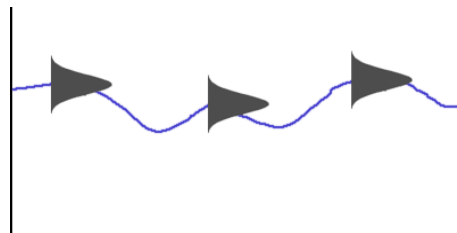


Abbildung 7: „Fortführungsrichtung des Signals“ ist normalverteilt

In diesem Kapitel folgt nun ein ziemlich komplexes Beispiel für Signalüberlagerungen und die zur Rückgewinnung benötigten Filter. Meine Kurzbeschreibung: das vorhandene weisse Rauschen kann ein empfangenes Signal völlig überdecken. Unser System soll nun erkennen, ob es sich beim empfangenen Signal um ein reines Rauschen oder ein durch Rauschen überdecktes Signal handelt. Dazu kann man einen passenden Filter entwickeln, welcher die klare Detektion eines Signals erlaubt.

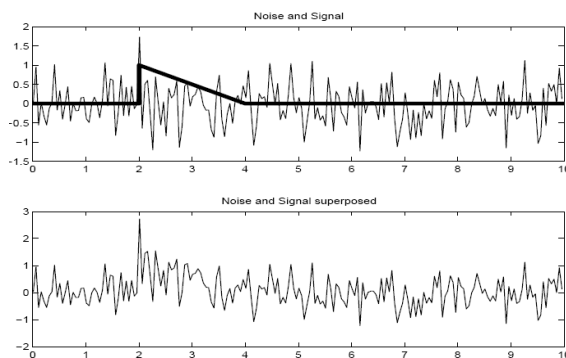


Abbildung 8: Signal nicht mehr erkennbar

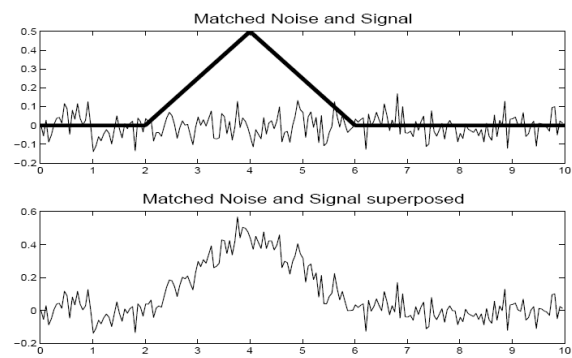
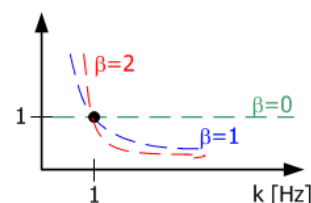


Abbildung 9: Mit geeignetem Filter Signal erkennbar

2.4. Farbiges Rauschen

Im Gegensatz zum weissen Rauschen besitzt das farbige Rauschen kein flaches Frequenzspektrum sondern enthält mehr Struktur. Die unterschiedlichen Farbnamen der sind von der Analogie des farbigen Lichts abgeleitet: würde man z.B. das „blaue Rauschen“ in Lichtwellen umwandeln, so würde ein blaues Licht entstehen.

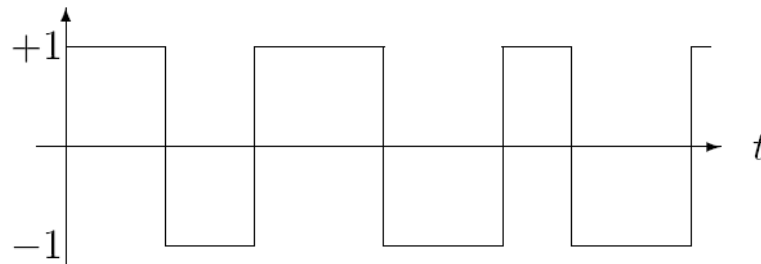
$$Frequenzspektrum = \frac{1}{k^\beta}$$



Abhängig vom jeweiligen Wert für β wird ein anderes „Rauschen“ unterschieden. Für $\beta=0$ erhalten wir weisses Rauschen, bei $\beta=1$ pinkes Rauschen und bei $\beta=2$ braunes Rauschen.

2.4.1. Der zufällige Telegraph Prozess

Dieser stochastische WSS-Prozess wechselt zwischen zwei Zuständen (+1 und -1). Es handelt sich um ein braunes Rauschen. Die Anzahl der 0-Durchgänge in einem bestimmten Zeitraum ist eine poissonverteilte Zufallsvariable mit dem Parameter $\lambda(t_2 - t_1)$.



Man kann nun die Wahrscheinlichkeit für die einzelnen Zustände zu jedem Zeitpunkt berechnen. Man erhält $P[X(t) = 1] = e^{-\lambda t} \cosh(\lambda t)$ und $P[X(t) = -1] = e^{-\lambda t} \sinh(\lambda t)$. Lassen wir t gegen unendlich laufen, so zeigt sich, dass beide Wahrscheinlichkeiten gerade 1/2 sind, was man intuitiv auch erwarten würde. Weiter kann nun noch die Autokorrelation R_x und die Power Spectral Density S_x berechnen. Diese lauten:

$$R(t_1, t_2) = e^{-\lambda|t_2 - t_1|} \qquad S_x(k) = \frac{2\lambda}{k^2 + \lambda^2}$$

2.4.2. Beispiel: Produktionsmaschine

Hierbei handelt es sich um eine Anwendung des braunen Rauschens. Stellen wir uns eine Maschine vor, welche eine **PRODUKTIONSRATE U** [Teile / Zeiteinheit] aufweist. Sowohl die Zeitdauer für den ordnungsgemässen Betrieb als auch die Wartungszeit sind exponentialverteilte Zufallsvariablen.

Durchschnittliche Laufzeit	$\frac{1}{\lambda}$
Durchschnittliche Ausfallzeit	$\frac{1}{\mu}$
Verhältnis Ausfall- zu Laufzeit	$I = \frac{1/\mu}{1/\lambda} = \frac{\lambda}{\mu}$
Durchschn. Produktionsrate	$m = \frac{\sum H}{H} = \frac{U}{I+1}$
Verteilung	$P(x \leq \sum H \leq x + dx) = \frac{1}{\sqrt{2\pi\sigma^2 H}} * \exp\left(-\frac{(x - mH)^2}{2\sigma^2 H}\right) dx$
Varianz der Verteilung	$\sigma^2 = \frac{2 * U^2 * I}{\mu(1+I)^3}$

Es handelt sich bei der Verteilung übrigens um eine Normalverteilung. Es kann somit bei Fragen zur Wahrscheinlichkeit, beispielsweise im 2-Sigabereich, die übliche Tabelle eingesetzt werden.

3. MARKOV KETTEN

3.1. Einleitung

Wir können einen stochastischen Prozess immer dann beschreiben, wenn wir alle möglichen Ausprägungen kennen. Das ist nicht immer einfach, deshalb wird für Markov-Prozesse eine Vereinfachung vorgenommen. Die **MARKOV EIGENSCHAFT** bedeutet, dass die Zukunft nur vom aktuellen Zustand, aber nicht von den vergangenen Eigenschaften abhängt.

	T diskret	T stetig
S diskret	zeitdiskrete Markov-Kette	zeitstetige Markov-Kette
S stetig	-	Diffusion Prozess

3.2. Zeitdiskrete Markov-Ketten

3.2.1. Grundlagen

Ein Beispiel für eine zeitdiskrete Markov-Kette ist der Random-Walk. Wie bereits erwähnt hängt die bedingte Wahrscheinlichkeit von X_{n+1} nur vom aktuellen Wert und in keinsten Weise von vergangenen Ereignissen ab (Stichwort: **HOMOGEN**). Dies drückt auch die Definition aus:

$$P(X_{n+1} = x_{n+1} | X_n = x_n, \dots, X_0 = x_0) = P(X_{n+1} = x_{n+1} | X_n = x_n)$$

Die Wahrscheinlichkeit, um mit einem Schritt von i nach j zu kommen, wird als **ÜBERGANGS-WAHRSCHEINLICHKEIT** bezeichnet.

$$p_{ij} = P(X_{n+1} = j | X_n = i)$$

Jegliche Übergänge können in einer **ÜBERGANGSMATRIX** festgehalten werden. Diese Übergangsmatrix enthält, mit Ausnahme der Startverteilung, alle Daten über den Prozess. Die Summe über jede Zeile entspricht dabei dem Wert 1.

$$P = \begin{pmatrix} p_{11} & p_{12} & p_{13} \\ p_{21} & p_{22} & p_{23} \\ p_{31} & p_{32} & p_{33} \end{pmatrix}$$

Die Übergangsmatrix kann auch als Zustandsdiagramm dargestellt werden. Nachfolgend ein Beispiel für eine 2x2 Matrix.

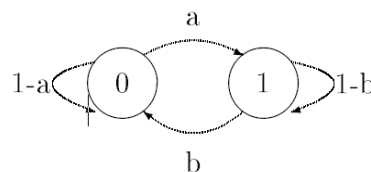


Abbildung 10: Beispiel für ein Zustandsdiagramm

Bei Zuständen gibt es zwei Spezialfälle: **ABSORBIERENDE** und **REFLEKTIERENDE** Zustände. Bei ersterem kommt man, sobald man den Zustand einmal betreten hat, nicht mehr weg. Es gibt also eine „Endlosschleife“ auf diesen Zustand. Die reflektierenden Zustände sind das genaue Gegenteil: mit einer Wahrscheinlichkeit von 1 wird vom betreffenden Zustand auf irgendeinen anderen Zustand gewechselt.

3.2.2. Chapman-Kolmogorov Gleichung

Bisher wurde immer nur ein „1-Schritt“-Übergang betrachtet. Bei der Chapman-Kolmogorov Gleichung werden nun komplexere Fälle betrachtet: wie gross ist die Wahrscheinlichkeit, um in m Schritten von Zustand i nach Zustand j zu kommen?

$$P_{ij}^{(m)} = P(X_m = j | X_0 = i)$$

Natürlich gibt es dafür mehr als eine Möglichkeit. Das Problem wird deshalb verallgemeinert: wir fügen einen Zwischenschritt ein und bewegen uns zuerst in n Schritten von i nach k und anschliessend in m Schritten von k zum Ziel j . Dies drückt die **CHAPMANN-KOLMOGOROV-GLEICHUNG** aus:

$$P_{ij}^{(n+m)} = \sum_k P_{ik}^{(n)} * P_{kj}^{(m)}$$

Dieser Vorgang vereinfacht sich nach einem komplexen Beweis auf das normale Matrixprodukt der Übergangsmatrix des stochastischen Prozesses.

$$P = \begin{pmatrix} 1/3 & 1/3 & 1/3 \\ 0 & 1/2 & 1/2 \\ 1 & 0 & 0 \end{pmatrix} \quad P^2 = \begin{pmatrix} 4/9 & 5/18 & 5/18 \\ 1/2 & 1/4 & 1/4 \\ 1/3 & 1/3 & 1/3 \end{pmatrix} \quad P^n \cong \begin{pmatrix} 0.428 & 0.285 & 0.285 \\ 0.428 & 0.285 & 0.285 \\ 0.428 & 0.285 & 0.285 \end{pmatrix}$$

Obiges Beispiel zeigt auch, dass für grosse n die Übergangsmatrix einen stabilen Zustand annehmen kann (**KONVERGENZMATRIX**). Dies muss allerdings nicht immer der Fall sein. Da alle 3 Zeilen identisch sind, beträgt die Wahrscheinlichkeit nach beliebig vielen Übergängen im Zustand 1 zu landen, genau 0.428 – unabhängig vom gewählten Startzustand!

3.2.3. Stationäre Verteilung

Eine homogene Markov-Kette ist stationär, wenn die Zustandsverteilung über der Zeit konstant ist. In diesem Fall ist die Startverteilung ein Eigenvektor der Übergangsmatrix.

$$\pi = \pi P$$

π ist die **STATIONÄRE VERTEILUNG** einer Kette und entspricht einer Sequenz $\pi=(\pi_0, \dots, \pi_n)$, deren Index durch den Zustandsraum $S=(0, \dots, n)$ vorgegeben ist, welche die folgenden Bedingungen erfüllt:

$$\pi_j = \sum_k \pi_k * P_{kj} \quad \sum_j \pi_j = 1$$

Auf ein konkretes Beispiel angewendet ergibt sich daraus nichts anderes, als ein lineares Gleichungssystem, welches aufgelöst werden kann. Die Werte der stationären Verteilung sagen aus, dass man (unabhängig vom Startzustand) nach genügend langer Zeit mit der betreffenden Wahrscheinlichkeit in diesem Zustand landet.

3.2.4. Erreichbare und kommunizierende Zustände, irreduzible Prozesse

- **ERREICHBAR:** Wenn mit endlich vielen Schritten ein Zustand j von i aus erreicht werden kann, so schreibt man $i \rightarrow j$.
- **KOMMUNIZIEREND:** Wenn die Zustände i und j sich gegenseitig „erreichen“ können, so spricht man von kommunizierenden Zuständen und schreibt $i \leftrightarrow j$.
- **IRREDUZIBEL:** Ein stochastischer Prozess ist dann irreduzibel, wenn jegliche Zustände untereinander kommunizierend sind. Eine irreduzible Markov-Kette hat keine oder eine stationäre Verteilung.

Nachfolgende Abbildung verdeutlicht diese Begriffe. Da sich Zustand 0 und Zustand 1 gegenseitig „erreichen“ können sind diese folglich „kommunizierend“. Zustände 1 und 2 können sich gegenseitig „nicht erreichen“. Der Prozess ist nicht irreduzibel, da nicht alle Zustände untereinander kommunizierend sind.

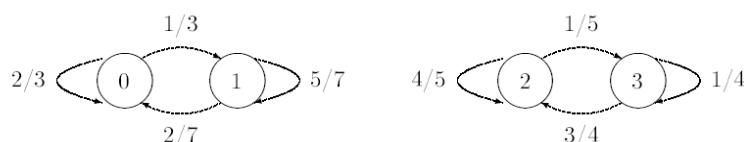


Abbildung 11: Beispiel für erreichbare, kommunizierende und irreduzible Zustände

3.2.5. Klassen

Alle untereinander kommunizierenden Zustände bilden eine **KLASSE**. In einer Markov Kette können mehrere disjunkte Klassen vorhanden sein, welche die Basis für komplizierte Ketten bilden:

$$S_X = C_1 \cup C_2 \cup \dots \cup C_N$$

Wenn eine Kette irreduzibel ist, so existiert gemäss Definition nur eine einzelne Klasse:

$$S_X = C_1$$

3.2.6. Rekurrente und transiente Zustände

Neben der Einteilung in Klassen kann unter den Zuständen eine weitere Unterteilung vorgenommen werden. Dazu braucht es wieder einige Begriffe:

f_i ist die Wahrscheinlichkeit, in den Zustand i zurück zu kehren, falls man in diesem begonnen hat.

- Ein Zustand i wird **REKURRENT** genannt, wenn $f_i = 1$. Wir kommen also garantiert irgendwann wieder zu diesem Zustand zurück. Ist eine Kette irreduzibel, so ist sie auch rekurrent.
- Ein Zustand i wird **TRANSIENT** genannt, wenn $f_i < 1$. Wir kommen ab einem bestimmten Zeitpunkt garantiert nie mehr zu diesem Zustand zurück.

Es können nicht alle Zustände einer Markov-Kette transient sein. Das ist intuitiv sofort klar, da andernfalls irgendwo ein Zustand vorhanden sein muss, in welchem man noch „flüchten“ könnte. Folglich muss mindestens einer der Zustände rekurrent sein.

- Ist ein Zustand rekurrent, so sind alle mit diesem Zustand kommunizierenden Zustände auch rekurrent.
- Ist ein Zustand transient, so sind alle mit diesem Zustand kommunizierenden Zustände auch transient.

3.2.7. Anzahl stationärer Verteilungen bestimmen

Es gibt Markov Ketten mit keiner, einer oder mehreren stationären Verteilungen.

- Eine irreduzible Markov-Kette hat maximale eine stationäre Verteilung (oder aber keine).
- Eine Markov-Kette, dessen Übergangsmatrix P^m für ein beliebiges m irgendwann überall positiv ist, besitzt genau eine stationäre Verteilung. Die Reihen von P konvergieren dabei gegen die stationäre Verteilung. Ein Beispiel einer solchen Kette findet sich in 3.2.2.

3.3. Random Walk auf Graphen

Ein Graph besteht aus einer Menge von Knoten (=Vertex V) und einer Menge von Kanten (=Edges E).

- **GRAD:** Der Grad eines Knotens ist die Anzahl der benachbarten Knoten.
- **VOLUMEN:** Das Volumen einer Teilmenge von Knoten ist die Summe über alle Grade der in dieser Teilmenge enthaltenen Knoten.
- **PFAD:** eine geordnete Sequenz von unterschiedlichen (!) Knoten.
- **VERBUNDENER GRAPH:** es gibt zwischen jeglichen Knoten einen Pfad.

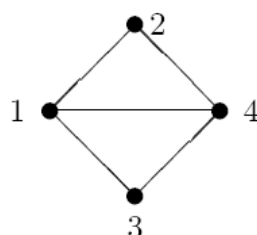


Abbildung 12: Bild eines Graphen (ohne Schleifen oder „doppelten Wege“)

Die **ADJAZENZ-MATRIX A** ist eine n*n-Matrix, welche mit 0en und 1en gefüllt ist. Eine 1 steht dabei für eine Kante zwischen zwei Knoten. Die **GRAD-MATRIX D** besitzt auf einer Diagonalen die Grade aller Knoten. Die **ÜBERGANGSMATRIX P** ist „fair verteilt“: besitzt ein Knoten drei Nachbarn (= Grad 3), so erhält jede wegführende Kante die Wahrscheinlichkeit $p = 1 / \text{Grad} = 1/3$.

$$A = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix} \quad D = \begin{pmatrix} 3 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 \\ 0 & 0 & 2 & 0 \\ 0 & 0 & 0 & 3 \end{pmatrix} \quad P = \begin{pmatrix} 0 & 1/3 & 1/3 & 1/3 \\ 1/2 & 0 & 0 & 1/2 \\ 1/2 & 0 & 0 & 1/2 \\ 1/3 & 1/3 & 1/3 & 0 \end{pmatrix}$$

Graphen haben grosse Ähnlichkeiten zu Markov-Ketten. Wenn ein Graph verbunden ist, so entspricht dies einer irreduziblen Kette, welche maximal eine stationäre Verteilung besitzt. Für den Random-Walk auf Graphen kann die stationäre Verteilung direkt mittels Formel bestimmt werden:

$$\pi_i = \frac{\text{Grad des Knotens}}{\text{Volumen des Graphen}} = \frac{d_i}{\sum_j d_j}$$

Oft interessiert man sich dafür, wie schnell eine Kette gegen die stationäre Verteilung konvergiert. Für den Random-Walk auf Graphen konvergiert die Matrix exponentiell schnell gegen diesen stationären Zustand.

$$\|vP^s - \pi\| \leq e^{-s\lambda} * \frac{\max_x(\sqrt{d_i})}{\max_y(\sqrt{d_j})} \quad \text{mit} \quad \lambda = \min(\lambda_2, 2 - \lambda_n)$$

λ_2 steht für den zweitgrössten und λ_n für den kleinsten Eigenwert (=Eigenvektor) von P.

3.3.1. Anwendungsbeispiel für Graphen: Bild-Segmentierung

Ein Anwendungsbeispiel für Graphen findet sich in der Bild-Segmentierung. Eine Grafik besteht üblicherweise aus einer Menge Pixeln. Diese Pixel werden in eine „reellwertige“ Adjazenzmatrix abgelegt. Die Zelle (i,j) steht dabei für eine Ähnlichkeit des Pixels i mit dem Pixel j. Wie diese Ähnlichkeit definiert wird ist dem Programmierer überlassen – oft verwendet werden Helligkeiten und Farbtöne.

Auf diese Matrix wird nun der **NCUT-ALGORITHMUS** angewendet. Dieser Algorithmus probiert die vorhandene Menge in zwei möglichst autonome Bereiche aufzuteilen. Autonom bedeutet in diesem Fall, dass zwar klar unterscheidbare Teile gebildet werden, welche in sich drin eine starke Verkettung besitzen, zum Gegenstück hin allerdings nur sehr wenige Verbindungen aufweisen.

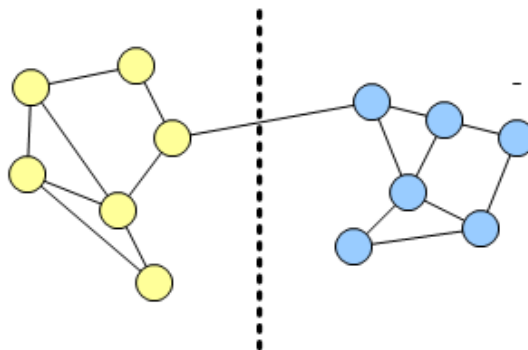


Abbildung 13: Skizzenhaftes Beispiel für eine Trennung der Gesamtmenge

Der Ncut-Algorithmus kann nach der Trennung selbstverständlich erneut auf die entstandenen Mengen angewendet werden.

3.4. Hidden Markov Model

Bei den normalen Markov-Modellen gehen wir davon aus, dass der aktuelle Zustand des Prozesses direkt erkenntlich ist. Bei den versteckten Markov-Modellen ist der Zustand des Prozesses nicht direkt ersichtlich. Dafür können wir die Ausgabe des Systems zu einem bestimmten Zeitpunkt beobachten. Jede diese Beobachtungen kann mit einer bestimmten Wahrscheinlichkeit von einem Zustand produziert worden sein – es handelt sich um eine ganz normale bedingte Wahrscheinlichkeit.

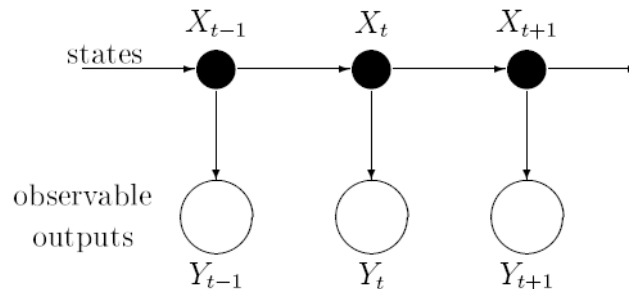


Abbildung 14: Beispiel einer HMM – nur die Outputs Y sind beobachtbar

HMM werden meist für Mustererkennungsaufgaben, beispielsweise die Sprach- oder Handschrift-erkennung eingesetzt. Ein einfaches Beispiel sind die Verkaufsdaten eines Eisverkäufers: sind diese in einer Tabelle gelistet, so können wir die Verkäufe direkt beobachten und probieren einen Rückschluss auf das jeweilige Tageswetter zu ziehen. Das Wetter entspricht dabei dem HMM.

Ein Hidden Markov Model (HMM) mit dem Zustandsraum $S = \{1, \dots, N\}$ und dem beobachtbaren Output $Y = \{s_1, \dots, s_M\}$ besteht aus drei Elementen:

Übergangsmatrix P	Bei P handelt es sich um eine gewöhnliche Übergangsmatrix einer Markov-Kette. Die Markov-Kette ist allerdings nicht direkt beobachtbar.
Ausgabe-W'keiten O	Diese $(N \times M)$ -Matrix enthält die Wahrscheinlichkeiten, dass in einem Zustand ein mögliche Beobachtung „produziert“ wird.
Startverteilung Π	Pro Zustand der Markov-Kette enthält die Startverteilung die Wahrscheinlichkeit, dass in diesem Zustand begonnen wird.

Bei HMM kämpft man mit drei Fundamentalen Problemen. Glücklicherweise existiert für jedes dieser drei Probleme ein passender Algorithmus, welche nachfolgend genauer betrachtet werden.

3.4.1. Forward-Backward Algorithmus

Angenommen, dass wir alle Parameter (P, O, Π) kennen, soll die Wahrscheinlichkeit für eine bestimmte Ausgabesequenz S berechnet werden. Weiter soll die Wahrscheinlichkeit berechnet werden, dass die Markov-Kette sich in einem Zustand befindet, wissend, dass wir die Ausgabesequenz (s_1, s_2, \dots, s_N) beobachtet haben.

$$P(Y = s_1, s_2, \dots, s_n) = ? \quad P(X_n | s_1, s_2, \dots, s_n) = ?$$

Diese Fragen können mit Hilfe des Forward-Backward Algorithmus beantwortet werden. Der FW-Algorithmus besteht aus zwei rekursiven Berechnungsformeln: einer Forward- und einer Backward-Gleichung.

Forward-Gleichung	Backward-Gleichung
$F_1(j) = \Pi_j * p(s_1 j)$ $F_n(j) = p(s_n j) * \sum_{i \in S} F_{n-1}(i) * p_{ij}$	$B_{n-1}(i) = \sum_{j \in S} p_{ij} * p(s_n j)$ $B_k(i) = \sum_{j \in S} p(s_{k+1} j) * p_{ij} * B_{k+1}(j)$

- s_n steht für die ersichtliche Beobachtung zum Zeitpunkt n
- j steht für den Zustand der versteckten Markov-Kette HMM
- Π_j ist die Anfangswahrscheinlichkeit für den Zustand j
- $p(s_n | j)$ beschreibt die Wahrscheinlichkeit, dass die Beobachtung s_n im Zustand j produziert wird
- p_{ij} ist die Übergangswahrscheinlichkeit, um in einem Schritt von i nach j zu wechseln.
- $F_n(j)$ steht für die W'keit, dass zum Zeitpunkt n die HMM im Zustand j ist.
- $B_k(j)$ steht für die W'keit, die zukünftige Sequenz zu beobachten, wissend, im Zustand j zu sein.

Mit Hilfe der Forward-Gleichung können nun die Wahrscheinlichkeiten für die in der Einleitung beschriebenen Probleme berechnen sowie den Zustand j bei Beobachtung s_n bestimmen:

W'keit für Zustandsfolge	W'keit für Zustand j bei Beobachtung s_n
$P(Y = s_1, \dots, s_n) = \sum_{i \in S} F_n(i)$	$P(X_n = j Y^n = s_n) = \frac{F_n(j)}{\sum_{i \in S} F_n(i)}$

Um die Berechnungsgeschwindigkeit zu erhöhen können beide Varianten parallel eingesetzt werden. In diesem Fall spricht man von der **FORWARD-BACKWARD-GLEICHUNG**.

$$P(Y^n = s_1, \dots, s_n) = \sum_{j \in S} F_k(j) * B_k(j)$$

Sind also beide Rekursionen beim Zeitpunkt k angelangt, so kann die Berechnung gestoppt und obige Formel angewendet werden.

Sowohl die Berechnungen des Forward- als auch diejenigen des Backward-Algorithmus führen sehr schnell zu kleinen Zahlen. Bei langen Beobachtungssequenzen werden die berechneten Werte so klein, dass der Algorithmus wegen Genauigkeitsproblemen fehlschlägt. Um dies zu vermeiden, können die Zwischenschritte auf jeder Stufe so skaliert werden, dass die Summe aller Teilwahrscheinlichkeiten die Zahl 1 ergibt. Man spricht von einem so genannten **SCALING**.

$$sc_k = \sum_{i \in S} F_k(i) \quad SF_k(j) = \frac{F_k(j)}{sc_k} \quad \sum_{i \in S} SF_k(i) = 1$$

Die Berechnung für die Wahrscheinlichkeit einer bestimmten Zustandsfolge berechnet sich beim Scaling über die Multiplikation der einzelnen Scalingfaktoren sc :

$$P(Y^n = s_1, \dots, s_n) = \prod_k sc_k$$

Die Multiplikation der Skalierungsfaktoren sc ist für viele Beobachtungen problematisch, da die Zahl gegen 0 tendiert. Aus diesem Fall wird oft über jegliche Werte der Logarithmus genommen:

$$\ln(P(Y^n = s_1, \dots, s_n)) = \sum_k \ln(sc_k)$$

Selbstverständlich können auch die Zwischenwerte B_k des Backward-Algorithmus skaliert werden. Das Verfahren ist in diesem Fall anaolg den oben beschriebenen Methoden.

3.4.2. Viterbi-Algorithmus

Angenommen, dass wir alle Parameter (P, O, Π) kennen, soll diejenige Sequenz von versteckten Zuständen X bestimmt werden, welche am wahrscheinlichsten die beobachtete Ausgabesequenz S produziert haben könnte. Viterbi findet garantiert die mathematisch höchste Wahrscheinlichkeit. Diese Wahrscheinlichkeit ist oft höher als die Wahrscheinlichkeit der realen Zustandssequenz, welche den beobachteten Output produziert hat.

Wir unterscheiden zwischen zwei grundsätzlich unterschiedlichen Ansätzen: lokale und globale Optimierung. Bei der **LOKALEN OPTIMIERUNG** wird versucht wir die Anzahl der korrekt vorhergesagten Zustände möglichst gross zu bekommen. Dazu wird für jeden Zeitpunkt derjenige Zustand j gewählt, welcher die grösste Wahrscheinlichkeit $P(X_k = j | Y^n = s_n)$ aufweist.

Die **GLOBALE OPTIMIERUNG** betrachtet die Zustandssequenz als eine gemeinsame Einheit. Es wird versucht, diejenige Sequenz von versteckten Zuständen zu bestimmen, welche am wahrscheinlichsten die beobachtete Ausgabesequenz S produziert haben könnte. Diese Variante entspricht dem Viterbi-Algorithmus. Wie schon der Forward-Backward-Algorithmus kann auch dieser rekursiv implementiert werden.

$$V_1(j) = \Pi_j * p(s_1 | j) \qquad V_n(j) = p(s_n | j) * \max_{i \in S} (V_{n-1}(i) * p_{ij})$$

- j steht für den Zustand der versteckten Markov-Kette HMM
- Π_j ist die Anfangswahrscheinlichkeit für den Zustand j
- $p(s_n | j)$ beschreibt die Wahrscheinlichkeit, dass die Beobachtung s_n im Zustand j produziert wird
- p_{ij} ist die Übergangswahrscheinlichkeit, um in einem Schritt von i nach j zu wechseln.

Zum Bestimmen der wahrscheinlichsten Zustandsfolge müssen Hilfwerte berechnet werden:

$$i_n = \arg \max_{i \in S} [V_n(i)] \qquad i_k(j) = \arg \max_{i \in S} [p_{ij} * V_k(i)]$$

Hat man alle diese Hilfwerte bestimmt, so kann man, beginnend bei i_n , die wahrscheinlichste Zustandsfolge bestimmen:

$$X_n = i_n \qquad X_k = i_k(X_{k+1})$$

Wir beginnen also beim letzten Zeitpunkt und arbeiten uns schrittweise bis zum Zeitpunkt 1 nach vorne.

Wie bereits beim Forward-Backward-Algorithmus, führt auch der Viterbi-Algorithmus bei langen Beobachtungssequenzen zu sehr kleinen Zahlen. Darum sollten die berechneten so **SKALIERT** werden, dass ihre Summe pro Zeitpunkt jeweils 1 ergibt.

3.4.3. Baum-Welch-Algorithmus

Ausgehend von einer Ausgabesequenz S bestimmt der Baum-Welch-Algorithmus die bestmöglichen Parameter (P, O, Π) eines unbekanntes HMM Modells. Es handelt sich beim BW-Algorithmus um ein Optimierungsproblem. Der Algorithmus läuft in Iterationen ab. In jeder Iteration nähert man sich mehr der optimalen Parameterwahl an. Der Algorithmus endet, sobald die „Verbesserung der Parameter“ zwischen zwei Schritt ein vorher festgelegtes Signifikanzniveau unterschreitet. Innerhalb der Iterationen werden die aus dem Kapitel 3.4.1 bekannten Forward- und Backward-Werte berechnet.

3.4.4. Beispiel: Produktionsmaschine

Stellen wir uns eine Maschine vor, welche zwischen einem guten und einem schlechten Produktionszustand (= G, S) wechseln kann. Befindet sich die Maschine einmal im schlechten Zustand, so kann sie nicht mehr in den guten Zustand zurück wechseln. Der Zustand der Maschine ist von aussen nicht ersichtlich.

In beiden Zuständen produziert die Maschine entweder akzeptable oder inakzeptable Teile (= A, I) welche beobachtet werden können. Die Wahrscheinlichkeit, dass im guten Zustand ein akzeptables Teil produziert wird beträgt 0.99, im schlechten Zustand nur 0.96.

$$P = \begin{pmatrix} p_{GG} & p_{GS} \\ p_{SG} & p_{SS} \end{pmatrix} = \begin{pmatrix} 0.9 & 0.1 \\ 0 & 1 \end{pmatrix} \quad D = \begin{pmatrix} p(A|G) & p(I|G) \\ p(A|S) & p(I|S) \end{pmatrix} = \begin{pmatrix} 0.99 & 0.01 \\ 0.96 & 0.04 \end{pmatrix} \quad P = (\Pi_G \quad \Pi_S) = (0.8 \quad 0.2)$$

Nehmen wir nun an, dass wir die Folge $S_3 = (A, I, A)$ beobachten. Folgende Fragen treten auf:

- 1) Wie gross ist die Wahrscheinlichkeit für die Folge S_3 (Forward-Algorithmus)?
- 2) Wie gross ist die Wahrscheinlichkeit für die Folge S_3 (Forward-Algorithmus mit Skalierung)?
- 3) Wie gross ist die Wahrscheinlichkeit für die Folge S_3 (Forward- und Backward-Algorithmus)?
- 4) Wie gross ist die Wahrscheinlichkeit, dass das 3. Produkt im guten Zustand G produziert wurde?
- 5) Wie gross ist die Wahrscheinlichkeit, dass das nächste Produkt im Zustand G produziert wird?
- 6) Wie gross ist die Wahrscheinlichkeit, dass das nächste Produkt akzeptabel ist?
- 7) Welches ist die wahrscheinlichste Zustandssequenz für die Folge $S_3 = (A, I, A)$?

1) Wahrscheinlichkeit für die Beobachtungsfolge (A,I,A) mit FW-Algorithmus:

$$F_1(G) = \Pi_G * p(A|G) = 0.8 * 0.99 = \underline{0.792}$$

$$F_1(S) = \Pi_S * p(A|S) = 0.2 * 0.96 = \underline{0.192}$$

$$F_2(G) = p(I|G) * [F_1(G) * p_{GG} + F_1(S) * p_{SG}] = 0.01 * [0.792 * 0.9 + 0.192 * 0] = \underline{0.00713}$$

$$F_2(S) = p(I|S) * [F_1(G) * p_{GS} + F_1(S) * p_{SS}] = 0.04 * [0.792 * 0.1 + 0.192 * 1] = \underline{0.01085}$$

$$F_3(G) = p(A|G) * [F_2(G) * p_{GG} + F_2(S) * p_{SG}] = 0.99 * [0.00713 * 0.9 + 0.01085 * 0] = \underline{0.00635}$$

$$F_3(S) = p(A|S) * [F_2(G) * p_{GS} + F_2(S) * p_{SS}] = 0.96 * [0.00713 * 0.1 + 0.01085 * 1] = \underline{0.011008}$$

$$P(S = (A, I, A)) = F_3(G) + F_3(S) = 0.006351 + 0.011008 = \underline{\underline{0.0174}}$$

2) Wahrscheinlichkeit für die Beobachtungsfolge (A,I,A) mit Skalierung:

$$F_1(G) = \Pi_G * p(A|G) = 0.8 * 0.99 = \underline{0.792}$$

$$F_1(S) = \Pi_S * p(A|S) = 0.2 * 0.96 = \underline{0.192}$$

$$sc_1 = F_1(G) + F_1(S) = 0.792 + 0.192 = \underline{0.984}$$

$$SF_1(G) = F_1(G) / sc_1 = 0.792 / 0.984 = \underline{0.805}$$

$$SF_1(S) = F_1(S) / sc_1 = 0.192 / 0.984 = \underline{0.195}$$

$$F_2(G) = p(I|G) * [SF_1(G) * p_{GG} + SF_1(S) * p_{SG}] = 0.01 * [0.805 * 0.9 + 0.195 * 0] = \underline{0.00725}$$

$$F_2(S) = p(I|S) * [SF_1(G) * p_{GS} + SF_1(S) * p_{SS}] = 0.04 * [0.805 * 0.1 + 0.195 * 1] = \underline{0.01102}$$

$$sc_2 = F_2(G) + F_2(S) = 0.00725 + 0.01102 = \underline{0.01827}$$

$$SF_2(G) = F_2(G) / sc_2 = 0.00725 / 0.01827 = \underline{0.396}$$

$$SF_2(S) = F_2(S) / sc_2 = 0.01102 / 0.01827 = \underline{0.604}$$

$$F_3(G) = p(A|G) * [SF_2(G) * p_{GG} + SF_2(S) * p_{SG}] = 0.99 * [0.396 * 0.9 + 0.604 * 0] = \underline{0.3528}$$

$$F_3(S) = p(A|S) * [SF_2(G) * p_{GS} + SF_2(S) * p_{SS}] = 0.96 * [0.396 * 0.1 + 0.604 * 1] = \underline{0.6178}$$

$$sc_3 = F_3(G) + F_3(S) = 0.3528 + 0.6178 = \underline{0.9707}$$

$$SF_3(G) = F_3(G) / sc_3 = 0.3528 / 0.9707 = \underline{0.364}$$

$$SF_3(S) = F_3(S) / sc_3 = 0.6178 / 0.9707 = \underline{0.636}$$

$$P(S = (A, I, A)) = sc_1 * sc_2 * sc_3 = 0.984 * 0.01827 * 0.9707 = \underline{0.0174}$$

3) Wahrscheinlichkeit für die Beobachtungsfolge (A, I, A) mit FW/BW-Algorithmus:

$$F_1(G) = \Pi_G * p(A|G) = 0.8 * 0.99 = \underline{0.792}$$

$$F_1(S) = \Pi_S * p(A|S) = 0.2 * 0.96 = \underline{0.192}$$

$$F_2(G) = p(I|G) * [F_1(G) * p_{GG} + F_1(S) * p_{SG}] = 0.01 * [0.792 * 0.9 + 0.192 * 0] = \underline{0.00713}$$

$$F_2(S) = p(I|S) * [F_1(G) * p_{GS} + F_1(S) * p_{SS}] = 0.04 * [0.792 * 0.1 + 0.192 * 1] = \underline{0.01085}$$

$$B_2(G) = p_{GG} * p(A|G) + p_{GS} * p(A|S) = 0.9 * 0.99 + 0.1 * 0.96 = \underline{0.987}$$

$$B_2(S) = p_{SG} * p(A|G) + p_{SS} * p(A|S) = 0 * 0.99 + 1 * 0.96 = \underline{0.96}$$

$$B_1(G) = p(I|G) * p_{GG} * B_2(G) + p(I|S) * p_{GS} * B_2(S) = 0.01 * 0.9 * 0.987 + 0.04 * 0.1 * 0.96 = \underline{0.013}$$

$$B_1(S) = p(I|G) * p_{SG} * B_2(G) + p(I|S) * p_{SS} * B_2(S) = 0.01 * 0 * 0.987 + 0.04 * 1 * 0.96 = \underline{0.0384}$$

$$P(S = (A, I, A)) = F_2(G) * B_2(G) + F_2(S) * B_2(S) = 0.00713 * 0.987 + 0.01085 * 0.96 = \underline{0.0174}$$

$$P(S = (A, I, A)) = F_1(G) * B_1(G) + F_1(S) * B_1(S) = 0.792 * 0.013 + 0.192 * 0.0384 = \underline{0.0174}$$

4) Wahrscheinlichkeit, dass 3. Produkt im guten Zustand produziert wurde:

$$P(X_3 = G | (A, I, A)) = \frac{F_3(G)}{F_3(G) + F_3(S)} = \frac{0.006351}{0.006351 + 0.011008} = \underline{0.3639}$$

5) Wahrscheinlichkeit, dass nächstes Produkt im Zustand G produziert wird:

$$\begin{aligned} P(X_4 = G | (A, I, A)) &= P(X_4 = G | (A, I, A), X_3 = G) * P(X_3 = G | (A, I, A)) + \\ &\quad P(X_4 = G | (A, I, A), X_3 = S) * P(X_3 = S | (A, I, A)) \\ &= P(X_4 = G | X_3 = G) * P(X_3 = G | (A, I, A)) + \\ &\quad P(X_4 = G | X_3 = S) * P(X_3 = S | (A, I, A)) \\ &= p_{GG} * P(X_3 = G | (A, I, A)) + p_{GS} * P(X_3 = S | (A, I, A)) \\ &= 0.9 * 0.364 + 0 * (1 - 0.364) = \underline{0.3276} \end{aligned}$$

6) Wahrscheinlichkeit, dass nächstes Produkt akzeptabel ist:

$$\begin{aligned} P(S_4 = A | (A, I, A)) &= P(S_4 = A | (A, I, A), X_4 = G) * P(X_4 = G | (A, I, A)) + \\ &\quad P(S_4 = A | (A, I, A), X_4 = S) * P(X_4 = S | (A, I, A)) \\ &= P(S_4 = A | X_4 = G) * P(X_4 = G | (A, I, A)) + \\ &\quad P(S_4 = A | X_4 = S) * P(X_4 = S | (A, I, A)) \\ &= P(A|G) * P(X_4 = G | (A, I, A)) + p(A|S) * P(X_4 = S | (A, I, A)) \\ &= 0.99 * 0.3276 + 0.96 * (1 - 0.3276) = \underline{0.969} \end{aligned}$$

7) Wahrscheinlichste Zustandssequenz bestimmen (Viterbi-Algorithmus):

$$V_1(G) = \Pi_G * p(A|G) = 0.8 * 0.99 = \underline{0.792}$$

$$V_1(S) = \Pi_S * p(A|S) = 0.2 * 0.96 = \underline{0.192}$$

$$V_2(G) = p(I|G) * \max[V_1(G) * p_{GG}, V_1(S) * p_{SG}] = 0.01 * \max[0.792 * 0.9, 0.192 * 0] = \underline{0.0071}$$

$$V_2(S) = p(I|S) * \max[V_1(G) * p_{GS}, V_1(S) * p_{SS}] = 0.04 * \max[0.792 * 0.1, 0.192 * 1] = \underline{0.0076}$$

$$V_3(G) = p(A|G) * \max[V_2(G) * p_{GG}, V_2(S) * p_{SG}] = 0.99 * \max[0.0071 * 0.9, 0.0076 * 0] = \underline{0.0064}$$

$$V_3(S) = p(A|S) * \max[V_2(G) * p_{GS}, V_2(S) * p_{SS}] = 0.96 * \max[0.0071 * 0.1, 0.0076 * 1] = \underline{0.0074}$$

$$i_3 = \arg \max[V_3(G), V_3(S)] = \arg \max[0.0064, 0.0074] = \underline{S}$$

$$i_2(G) = \arg \max[p_{GG} * V_2(G), p_{SG} * V_2(S)] = \arg \max[0.9 * 0.0071, 0 * 0.0076] = \underline{G}$$

$$i_2(S) = \arg \max[p_{GS} * V_2(G), p_{SS} * V_2(S)] = \arg \max[0.1 * 0.0071, 1 * 0.0076] = \underline{S}$$

$$i_1(G) = \arg \max[p_{GG} * V_1(G), p_{SG} * V_1(S)] = \arg \max[0.9 * 0.792, 0 * 0.192] = \underline{G}$$

$$i_1(S) = \arg \max[p_{GS} * V_1(G), p_{SS} * V_1(S)] = \arg \max[0.1 * 0.792, 1 * 0.192] = \underline{S}$$

$$X_3 = i_3 = \underline{S}$$

$$X_2 = i_2(X_3) = i_2(S) = \underline{S}$$

$$X_1 = i_1(X_2) = i_1(S) = \underline{S}$$

3.5. Hidden Markov Models – Spracherkennung

Bei der Spracherkennung handelt es sich um eine praktische Anwendung für versteckte Markov Ketten (HMM). Zur Erstellung eines erfolgreichen Spracherkenners erstellt sind verschiedene Teilschritte nötig.

3.5.1. Schritt 1 – Erstellen des Codebooks

Im Codebook sind die Grundbausteine („Alphabet“) der Spracherkennung enthalten. Zur Erstellung des Codebooks werden möglichst viele gesprochene Wörter, sowohl von Männern als auch Frauen, analysiert. Dieser Analysevorgang wird auch **FEATUREEXTRACTION** genannt. Dabei wird zuerst jedes Wort in (teilweise überlappende) Frames einer bestimmten Zeitdauer aufgeteilt. Anschließend werden für jeden Frame bestimmte Kennzahlen („Features“) berechnet. Dabei existiert eine Vielzahl möglicher Kennwerte. In unserem Beispiel werden $M=24$ Features berücksichtigt. Am gebräuchlichsten ist die Verwendung von $M=39$ Kennwerten.

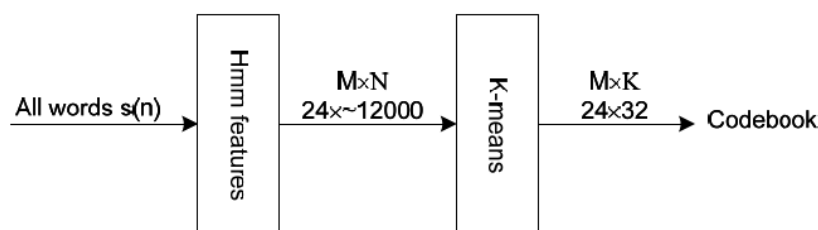


Abbildung 15: Schematische Erstellung des Codebooks

Optimal wäre es nun natürlich, wenn man jeden dieser Frames mit seinen Features zur Erkennung verwenden könnte. Aus Gründen des Speicherplatzes und der Laufzeit der Erkennung ist dies jedoch nicht möglich. Darum wird jedem vorhandenen Frame (über 10'000!) mittels K-Means Clustering eines von $K=32$ repräsentativen Frames zugewiesen. Diese repräsentativen Frames entsprechen dem Codebook. Der Wert 32 stellt dabei eine untere Grenze dar, welche für ein kleines zu erkennendes Vokabular genügt. Grössere Systeme verwenden bis zu 256 representative Frames.

3.5.2. Schritt 2 – Trainieren der Modelle

In diesem Schritt wird für jedes der zu erkennenden Worte ein eigenes Hidden Markov Model erstellt. Dazu werden die zu erkennenden Wörter von unterschiedlichen Testpersonen ausgesprochen. Auf diese Sprachdaten wird nun wiederum eine Featureextraktion (siehe Kapitel 3.5.1) durchgeführt. Jedes der erkannten Frames wird auf eines der $K=32$ repräsentativen Frames im Codebook abgebildet. Folglich erhalten wir pro gesprochenes Wort eine Liste der jeweiligen Frames. Diese Liste verwenden wir nun als die Beobachtung einer versteckten Markov-Kette.

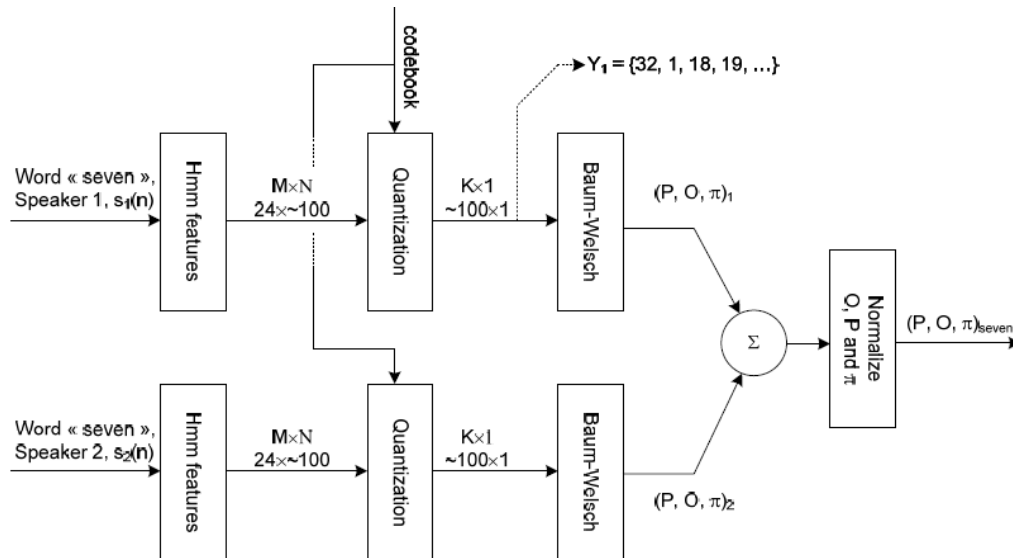


Abbildung 16: Schematische Darstellung des Training-Vorgangs

Über die Liste von Beobachtungen können wir nun mittels des Baum-Welsh-Algorithmus die Parameter (P, O, Π) der versteckten Markov-Kette berechnen lassen. Dabei können wir festlegen, wie viele interne Zustände S unsere Markov-Kette besitzen soll (in unserem Beispiel ist $S=5$). Da wir jedes Wort mit mehreren Personen trainieren lassen, müssen zum Schluss alle berechneten Modelle pro zu erkennendem Wort miteinander kombiniert werden (Durchschnitt). Als Resultat des Trainingsvorgangs erhalten wir pro zu erkennendem Wort eine eigene HMM.

3.5.3. Schritt 3 – Erkennen der Wörter

Die Schritte 1 und 2 werden selten ausgeführt, die benötigte Rechenzeit ist somit vernachlässigbar. Ganz anders der Erkennungsvorgang: dieser muss so schnell als möglich durchgeführt werden können. Folglich ist der Erkennungsvorgang auch nicht besonders komplex: für das zu erkennende Wort werden wiederum die Charakteristiken für jedes einzelne Frame berechnet. Jedes dieser Frames wird darauf einem der $K=32$ repräsentativen Frames des Codebooks zugewiesen – wir erhalten als Resultat eine „Beobachtungssequenz“. Nun wird über den Forward-Algorithmus für jedes HMM (pro Wort ein HMM) die Wahrscheinlichkeit für diese Beobachtungssequenz berechnet. Dabei sollte das HMM, welches dem zu erkennenden Wort entspricht, die höchste Wahrscheinlichkeit produzieren.

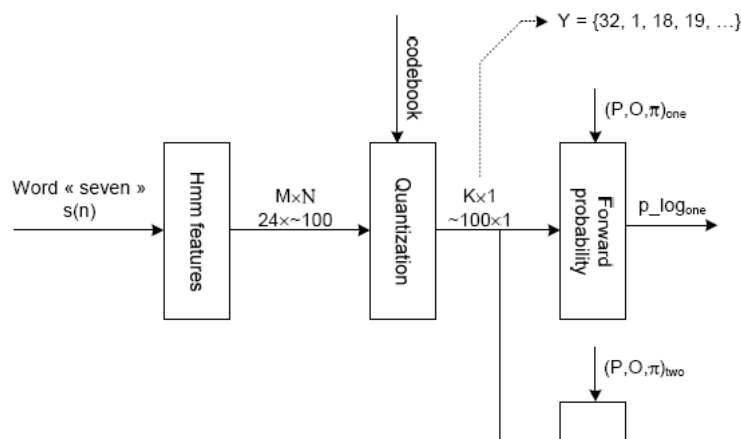


Abbildung 17: Schematische Darstellung des Erkennungsvorgangs

3.5.4. Verbesserungsmöglichkeiten

- Trainieren der Modelle mit mehr Testpersonen (mindestens 10, besser 100)
- Vorverarbeitung des Signals (Entfernen von Bereichen ohne Geräusche)
- Die Zahl der Charakteristiken M eines Frames erhöhen. In unserem Beispiel betrug M=24. Bei modernen Systemen werden M=39 Kennwerte eingesetzt.
- **CONTINUOUS DENSITY HMM:** im Gegensatz zur Vektor-Quantifizierung (VQ) wird bei diesem Ansatz eine stetige Funktion („Normalverteilung“) verwendet.
- **LEFT-RIGHT MODELS:** In unserem System sind die Markov-Ketten voll vermascht – jeder Zustand kann mit jedem kommunizieren. Bei Left-Right Models werden nur noch „Sprünge“ von 2 Zuständen erlaubt. Somit werden grosse Teile der Übergangsmatrix den Wert 0 aufweisen.

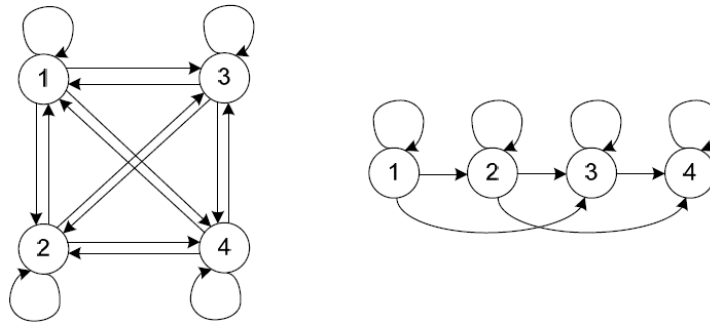


Abbildung 18: Ergodische und Left-Right Markovketten

3.5.5. Aktuelle Systeme

Alle aktuellen Systeme basieren auf Left-Right Models, entweder auf Vektor- oder Continuous Density Quantifizierung. Es werden M=39 Charakteristiken und K=32...256 repräsentative Frames eingesetzt. Die Anzahl an versteckten Zuständen der Markovketten schwankt zwischen S=2...10.

3.6. Zeitstetige Markovketten

3.6.1. Haltezeit

Bei den zeitdiskreten Markovketten haben wir keine Aussage über die in einem Zustand verbrachte Zeit gemacht. Bei zeitstetigen Markovketten folgt die in einem Zustand verbrachte Zeit, auch **HALTEZEIT** genannt, einer **EXPONENTIALVERTEILUNG** (siehe Kapitel 1.6).

Exponential RV ($X \sim \text{Exp}(\lambda)$)			
$P(X)$	$f(x) = \lambda e^{-\lambda x}$	$F(X)$	$F(x) = 1 - e^{-\lambda x}$
$E(X)$	$E(X) = \frac{1}{\lambda}$	$\text{VAR}(X)$	$\text{VAR}(X) = \frac{1}{\lambda^2}$
$\phi(t)$	$\phi(t) = \frac{\lambda}{\lambda - t}$		

Der Parameter λ kann als „Rate“ interpretiert werden. Weist dieser den Wert 2 auf, so kann alle 1/2 Zeiteinheiten (z.B. Sekunden) mit einem Ereignis gerechnet werden.

3.6.2. Memoryless Property

Die Exponentialverteilung ist die einzige gedächtnislose, stetige Verteilung (**MEMORYLESS PROPERTY**). Diese Eigenschaft sagt aus, dass die Wahrscheinlichkeit für ein Ereignis nicht von der bereits vergangenen Zeit abhängt.

$$P[X > s + t | X > t] = P[X > s]$$

Beispiel: Die Wahrscheinlichkeit, dass ein Mensch zum aktuellen Zeitpunkt für einen weiteren Zeitraum s lebt ist unabhängig vom bereits vergangenen (bzw. bereits gelebten) Zeitraum t.

3.6.3. Erlang Verteilung

Beobachten wir eine Sequenz von unabhängigen und identischen Exponentialverteilungen und summieren diese auf, so erhalten wir eine Zufallsvariable, die **ERLANG** verteilt ist.

$$f(x) = \frac{(\lambda x)^{n-1} * \lambda e^{-\lambda x}}{(n-1)!}$$

3.6.4. Eigenschaften einer zeitstetigen Markovkette

Wie bei den zeitdiskreten Markovketten hängt auch bei den zeitstetigen Markovketten die Wahrscheinlichkeit für den nächsten Zustand nur vom aktuellen Zustand ab, nicht aber von der gesamten Vorgeschichte.

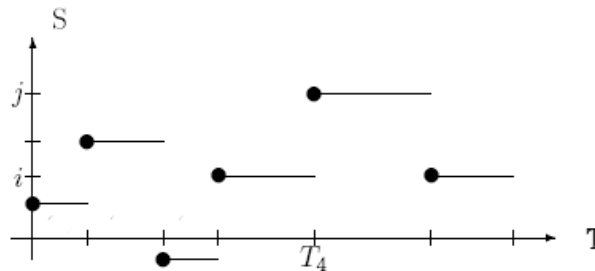


Abbildung 19: Skizze einer zeitstetigen Markovkette

Die Markovkette hat zeit-homogene (= unveränderlich über die Zeit) Übergangswahrscheinlichkeiten p_{ij} . Diese werden auch Übergangsfunktion genannt. Sie geben die Wahrscheinlichkeit, um im Zeitintervall $[t-s]$ vom Zustand i in den Zustand j zu wechseln.

$$p_{ij}(t, s) = p_{ij}(t - s) = P(X(t + s) = j | X(s) = i)$$

Jeder Zustand besitzt eine exponential verteilte und gedächtnislose **HALTEZEIT** τ_i mit Parameter ν_i . Das bedeutet, dass die Wahrscheinlichkeit noch länger in diesem Zustand zu verweilen unabhängig von der bereits vergangenen Zeit ist. Die vergangene Zeit vor einem Zustandwechsel beeinflusst in keiner Weise, auf welchen Zustand gewechselt werden soll. Allerdings muss in einen anderen Zustand gewechselt werden – der Prozess kann nicht in demselben Zustand verbleiben. Das bedeutet, dass die Übergangsmatrix auf einer Diagonalen nur 0en besitzt.

$$p_{ii} = 0 \quad \sum_{j \in S} p_{ij} = 1$$

3.6.5. Chapman-Kolmogorov

Wie bei den diskreten Markovketten existiert auch für die zeitstetigen Markovketten eine Chapman-Kolmogorov Gleichung. Diese drückt die Wahrscheinlichkeit für den Wechsel von Zustand i zum Zustand j im Zeitraum $[t-s]$ mit Zwischenhalt in Zustand k aus.

$$p_{ij}(t + s) = \sum_k p_{ik}(t) * p_{kj}(s)$$

Um die Übergangsfunktion einer Markovkette zu finden müsste die Chapman-Kolmogorov Gleichung gelöst werden. Das ist i. d. R. jedoch eine fast unmögliche Aufgabe. Glücklicherweise gibt es eine einfachere Variante unter Verwendung von **INSTANTANEOUS TRANSITION RATES**. Dozent Filliger bezeichnet diesen Wert auch als „Lust, von i nach j zu wechseln“. ν_i steht für den Parameter der exponentialverteilten Haltezeit eines Zustands.

$$\begin{aligned} \nu_{ij} &= \nu_i * p_{ij} \\ \nu_{ii} &= -\nu_i \end{aligned}$$

Schreiben wir alle diese Werte in eine Matrix, so erhalten wir die **GENERATING MATRIX**.

$$G = \begin{pmatrix} v_{11} & v_{12} & v_{13} \\ v_{21} & v_{22} & v_{23} \\ v_{31} & v_{32} & v_{33} \end{pmatrix}$$

Mit Hilfe der Generatormatrix und der **KOLMOGOROV BACKWARD EQUATION** kann ein Gleichungssystem von linearen Differentialgleichungen erstellt werden. Ein Beispiel dazu findet sich auf Seite 113 im Skript.

$$p'_{ij}(t) = \sum_{k \in S} v_{ik} * p_{kj}(t)$$

p_{ij} ist die Wahrscheinlichkeit, um im Zeitraum t von Zustand i zum Zustand j zu kommen. Die Zahl der zu erstellenden Gleichungen nimmt quadratisch mit der Anzahl der Zustände der Markovkette zu. Dies macht das Lösen der Differentialgleichungen sehr zeit- und rechenintensiv. Allerdings müssen wir das auch nicht können, noch einmal Glück gehabt! ☺

3.7. Zählprozess

Ein stochastischer Prozess wird Zählprozess genannt, wenn N_t eine Anzahl Vorkommnisse im Zeitraum t darstellt. Es gilt:

- $N_t \geq 0$, wobei N_t ein ganzzahliger Wert ist.
- Wenn $s < t$ dann $N_s < N_t$
- $N_t - N_s$ entspricht der Anzahl an Ereignissen im Zeitraum $(s, t]$



Abbildung 20: Zeitraum mit zwei Ereignissen

3.8. Poisson Prozess

Der Poisson Prozess ist die bekannteste zeitstetige Markovkette. Nachfolgend drei verschiedene Definitionen, welche alle identisch sind. Anwendungen findet der Prozess an verschiedensten Orten:

- Besucher eines Ladens
- Besucher einer Webseite
- Auftreten von Erdbeben (Anzahl)
- Eingehende Anrufe in einem Telefonzentrum
- Auftreten von Fehlern in einem Produktionsprozess

3.8.1. Definition 1 – Betrachtung der Poissonverteilung

Es handelt sich um einen Zählprozess mit $N_0=0$. Jeder Zustandswechsel erhöht diesen Wert um 1. Der Prozess hat **UNABHÄNGIGE INKREMENTE** (Kapitel 2.1.3). Das bedeutet, dass die in einem Zeitraum aufgetretenen Ereignisse die in einem anderen Zeitraum aufgetretenen Ereignisse nicht beeinflussen, solange sich diese Zeiträume nicht überlappen.

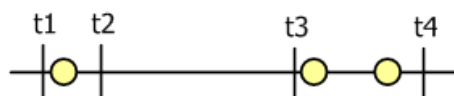


Abbildung 21: Zwei unabhängige Zeiträume, die sich gegenseitig nicht beeinflussen

Die wichtigste Eigenschaft dieser ersten Definition ist, dass die Anzahl Ereignisse in einem Zeitraum t einer **POISSONVERTEILUNG** (Kapitel 1.5) mit Parameter λt folgt:

$$P(N_{t+s} - N_s = n) = e^{-\lambda t} \frac{(\lambda t)^n}{n!}$$

Wichtig: Auch wenn die Anzahl Ereignisse einer Poissonverteilung folgt ist die Haltezeit in einem Zustand ist immernoch **EXPONENTIALVERTEILT** mit der Rate λ . In der Realität ist der Parameter λ oft unbekannt und muss anhand vorhandener Daten geschätzt werden. Eine Möglichkeit dazu ist:

$$\lambda = \frac{\text{AnzahlDerBeobachtungen}}{\text{Zeit}} = \frac{E(N_t)}{t}$$

Weiter kann über den Prozess gesagt werden, dass er **STATIONÄRE INKREMENTE** besitzt (Kapitel 2.1.3). Das bedeutet, dass der Prozess für alle Zeitabschnitte t dieselbe Verteilung besitzt. Allerdings ist der Prozess nicht stationär, weder **STRICT SENSE STATIONARY** noch **WIDE SENSE STATIONARY**. Das sehen wir daran, dass der Erwartungswert nicht konstant über die Zeit ist und die Autkovarianz nicht nur vom Zeitabschnitt $|t_1 - t_2|$ abhängt.

3.8.2. Definition 2 – Betrachtung des Verhaltens für kleine Zeitabschnitte

Die zweite Definition gibt, neben der Bedingung „Zählprozess“ und „stationäre Inkremente“, eine Schätzung für das Verhalten der Wahrscheinlichkeiten für kleine Zeitabschnitte.

$$P(N_t = 1) = \lambda t + o(h) \quad P(N_t \geq 2) = o(h)$$

Das bedeutet, dass für kleine Zeiträume die Wahrscheinlichkeit eines Ereignis proportional zur länge des Zeitraums ist. Die zweite Bedingung definiert, dass das Auftreten von zwei Ereignissen in einem kleinen Zeitraum gegen 0 tendiert.

3.8.3. Definition 3 – Betrachtung der exponentialverteilten Haltezeiten

Diese Definition betrachtet die Haltezeiten eines Poissonprozesses. Wie wir aus Kapitel 3.6 wissen sind die den Zuständen zugeordneten Haltezeiten τ eines zeitstetigen Prozesses exponentialverteilt (siehe Kapitel 3.6.1). Beim Poissonprozess besitzen alle Zustände dieselbe Haltezeit τ mit Parameter λ . Unter Berücksichtigung der **INSTANTENOUS TRANSITIONS RATES** (Kapitel 3.6.5) ergibt sich die folgende Generating Matrix für einen Poissonprozess:

$$P = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad G = \begin{pmatrix} -\lambda & \lambda & 0 & 0 \\ 0 & -\lambda & \lambda & 0 \\ 0 & 0 & -\lambda & \lambda \\ 0 & 0 & 0 & -\lambda \end{pmatrix}$$

3.8.4. Zusammenführen und Splitten

Betrachten wir zwei unabhängige Poissonprozesse mit den Parametern λ_1 und λ_2 , so können wir diese zu einem neuen Poissonprozess mit zusammenfassen:

$$\lambda = \lambda_1 + \lambda_2$$

Ebenso können wir einen einzelnen Poissonprozess in zwei einzelne Prozesse aufteilen. Dazu stellen wir uns vor, dass wir bei für jedes einzelne Vorkommnissen eine Münze werfen die angibt, ob das Element zum ersten oder zweiten Prozess gehören soll. Wir erhalten:

$$\lambda_1 = p\lambda \quad \lambda_2 = (1 - p)\lambda$$

3.8.5. Beispiele zum Poissonprozess

Seite 115 / Exa 48: Die Fehler einer Maschine folgen einem Poissonprozess mit Rate $\lambda = 2$ Ausfällen pro Woche. Im Intervall $[0,1]$ sind genau zwei Fehler aufgetreten. $t_0 > 3$ ist ein beliebiger Zeitpunkt.

- 1) Wkeit, das zum Zeitpunkt t_0 zwei Wochen seit dem letzten Fehler vergangen sind?
- 2) Wkeit, das zum Zeitpunkt t_0 zwei Wochen seit dem vorletzten Fehler vergangen sind?
- 3) Wkeit, das während den ersten zwei Tagen von t_0 kein Fehler auftritt?

1) Wkeit für 2 Wochen ohne Fehler seit letztem Fehler?

Die Zeiträume $[0,1]$ und $[t_0, t_{0-2}]$ überlappen sich nicht. Es gilt also:

$$P(N_{t_0} - N_{t_{0-2}} = 0 \mid N_1 - N_0 = 2) = P(N_{t_0} - N_{t_{0-2}} = 0) = e^{-2*2} * \frac{(2*2)^0}{0!} = \underline{\underline{e^{-4}}}$$

2) Wkeit für 2 Wochen seit vorletztem Fehler?

Wenn der vorletzte Fehler ausserhalb des Zeitraums $[t_0, t_{0-2}]$ liegen muss, so kann der letzte Fehler entweder auch ausserhalb des Zeitraums, oder aber innerhalb des Zeitraums aufgetreten sein. Die Zeiträume $[0,1]$ und $[t_0, t_{0-2}]$ überlappen sich auch hier nicht.

$$P(N_{t_0} - N_{t_{0-2}} \leq 1) = \left(e^{-2*2} * \frac{(2*2)^0}{0!} \right) + \left(e^{-2*2} * \frac{(2*2)^1}{1!} \right) = e^{-4} (1+4) = \underline{\underline{5e^{-4}}}$$

3) Wkeit für 2 Tage ohne Fehler?

Da die Rate $\lambda = 2$ für eine Woche gilt, müssen wir diese zuerst kürzen:

$$\lambda_{2d} = \lambda / 7 = \underline{\underline{2/7}}$$

Mit dieser Rate können wir jetzt die Wkeit für 0 Fehler berechnen:

$$P\left(N_{t_0 + \frac{2}{7}} - N_{t_0} = 0\right) = P\left(N_{\frac{2}{7}} = 0\right) = e^{-\frac{2}{7}*2} * \frac{\left(\frac{2}{7}*2\right)^0}{0!} = \underline{\underline{e^{-\frac{4}{7}}}}$$

Seite 118 / Exc 43: Kunden betreten ein Postamt gemäss einem Poissonprozess mit einer Rate von $\lambda = 2$ Kunden pro Minute.

- 1) Wkeit, das die Zeit zwischen dem zweiten und dritten Kunden grösser als 5 Minuten ist.
- 2) Wkeit, das die Zeit zwischen dem 10ten und 11ten Kunden kleiner als 1 Minute ist.

1) Wkeit, das Zeit zwischen 2. und 3. Kunden grösser als 5 Minuten ist?

Da die bereits eingetretenen Ereignisse einander nicht beeinflussen ist es uns eigentlich egal, ob es sich um welchen Kunden es sich handelt.

$$P(\tau > 5) = \int_5^{\infty} \lambda e^{-\lambda t} dt = -e^{-\lambda t} \Big|_{t=5}^{\infty} = -e^{-2*\infty} - (-e^{-2*5}) = 0 - (-e^{-10}) = \underline{\underline{e^{-10}}}$$

2) Wkeit, das Zeit zwischen 10. und 11. Kunden kleiner als 1 Minute ist?

$$P(\tau < 1) = \int_0^1 \lambda e^{-\lambda t} dt = 1 - e^{-\lambda t} \Big|_{t=0}^1 = (1 - e^{-2*1}) - (1 - e^{-2*0}) = \underline{\underline{1 - e^{-2}}}$$

Seite 119 / Exc 44: Die Verkehrsintensität auf einer Einbahnstrasse wird durch einen Poissonprozess mit der Intensität $\lambda = 1/6$ Autos pro Sekunde beschrieben. Ein Fussgänger benötigt 4 verkehrsfreie Sekunden um die Strasse zu überqueren.

- 1) Wkeit, das der Fussgänger warten muss.
- 2) Mittlere Länge des Zeitintervalls, in welchem der Fussgänger die Strasse passieren kann.
- 3) Mittlere Anzahl der Fahrzeuge, welche vor der Strassenüberquerung vorbeifahren.

1) Wkeit, das der Fussgänger warten muss?

$$P(\tau < 4) = \int_0^4 \lambda e^{-\lambda t} dt = 1 - e^{-\lambda t} \Big|_{t=0}^4 = (1 - e^{-1/6 \cdot 4}) - (1 - e^{-1/6 \cdot 0}) = \underline{\underline{1 - e^{-2/3}}}$$

2) Mittlere Länge des Zeitintervalls, welches eine Überquerung erlaubt?

Zur Lösung dieser Aufgabe müssen wir die Memoryless Property der exponentialverteilten Haltezeit berücksichtigen (Kapitel 3.6.2).

Der Erwartungswert der Haltezeit beträgt 6 Sekunden (= wir erwarten, dass durchschnittlich 6 Sekunden ohne Auto vergehen). Für eine Strassenüberquerung benötigt der Fussgänger 4 Sekunden. Zeitabschnitte, welche eine Überquerung erlauben sind somit im Durchschnitt $4 + 6 = 10$ s lang.

3) Mittlere Anzahl der Autos vor Überquerung?

Zur Lösung dieser Aufgabe muss man wissen, dass die Anzahl der eintreffenden Autos eine Zufallsvariable Y ist, welche einer geometrischen Verteilung (Kapitel 1.5) folgt. Der Parameter p dieser Verteilung ist die „Wahrscheinlichkeit für Eintreffen des erfolgreichen Ereignisses“. In unserem Fall ist das „erfolgreiche Ereignis“ das Überqueren der Strasse durch den Fussgänger.

$$p = P(\tau > 4) = \int_4^{\infty} \lambda e^{-\lambda t} dt = -e^{-\lambda t} \Big|_{t=4}^{\infty} = -e^{-1/6 \cdot \infty} - (-e^{-1/6 \cdot 4}) = 0 - (-e^{-4/6}) = \underline{\underline{e^{-2/3}}}$$

$$E(Y) = \frac{1}{p} = \frac{1}{e^{-2/3}} = \underline{\underline{e^{2/3}}}$$

3.9. Geburts- und Sterbeprozesse

Wenn wir den Poissonprozess so generalisieren, dass er neben dem Zählen nach oben auch nach unten zählen darf, so erhalten wir einen Geburts- und Sterbeprozess. Ein schönes Beispiel ist ein Postamt, bei welchem die Kunden ein Postamt betreten („Geburt“) und wieder verlassen („Sterben“). Sowohl die Zeitspanne („Haltezeit“) bis zur Ankuft, als auch die Zeitspanne bis zum Verlassen des Postamts ist exponentialverteilt. Die beiden Parameter der Exponentialverteilungen nennen wir Geburtsrate λ (für das Betreten) sowie Sterberate μ (für das Verlassen).

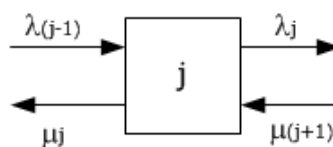


Abbildung 22: Skizze eines Zustands mit Geburts- und Sterberate

Unter Berücksichtigung von Kapitel 3.6.5 können die **INSTANTANEOUS TRANSITIONS RATES** bestimmt werden. Für diese gilt im Allgemeinen (siehe auch Kapitel 3.6.5):

$$v_{ii} = -v_i \qquad v_{ij} = v_i * p_{ij}$$

Die Geburts- und Sterberate kann auch zusammengefasst werden. Wir erhalten so für jeden Zustand den Parameter v_i der exponentialverteilten Haltezeiten τ_i . Diese Werte entsprechen der Diagonale der Generatormatrix (folgt unten).

$$v_0 = \lambda_0 \qquad v_i = \lambda_i + \mu_i \qquad v_n = \mu_n$$

Mit diesen Formeln erhalten wir für das Postamt-Beispiel mit 4 Warteplätzen in der Schlange die folgenden beiden Matrizen:

$$P = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 1-p & 0 & p & 0 \\ 0 & 1-p & 0 & p \\ 0 & 0 & 1 & 0 \end{pmatrix} \qquad G = \begin{pmatrix} -\lambda_0 & \lambda_0 & 0 & 0 \\ (1-p)\mu_1 & -(\lambda_1 + \mu_1) & p\lambda_1 & 0 \\ 0 & (1-p)\mu_2 & -(\lambda_2 + \mu_2) & p\lambda_2 \\ 0 & 0 & \mu_3 & -\mu_3 \end{pmatrix}$$

Mit Hilfe der Instantaneous Transition Rates und der anschliessenden resultierenden Kolmogorov Backward Gleichung erhalten wir wiederum zur Bestimmung der Übergangsfunktion ein Gleichungssystem mit mehreren Differentialgleichungen. Allerdings sind auch diese alles andere als einfach, wenn nicht gar unmöglich, zu lösen.

Einfacher ist es, das Verhalten für die Zeit nach unendlich zu untersuchen (**STATIONÄRE VERTEILUNG**). Pro Zustand j berechnen wir ein π_j , welches die Wahrscheinlichkeit für das Landen in diesem Zustand nach „unendlicher“ Zeit angibt. Die wichtigste Formel zur Bestimmung dieser Werte ist die Gleichgewichtsbedingung:

$$\pi_j v_j = \sum_{i \neq j} \pi_i v_{ij} \qquad \sum_{j=0}^{\infty} \pi_j = 1$$

Obige Formel heisst Gleichgewichtsbedingung, weil sie auf der linken Seite die „Lust zum Verlassen des Zustandes“ und auf der rechten Seite der Gleichung die „Lust zum Betreten des Zustandes“ besitzt. Nachfolgendes Beispiel soll das Aufstellen der Gleichungen verdeutlichen. Ein weiteres gutes Beispiel findet sich in der Aufgabe 14 der „Prüfungsvorbereitung“.

Beispiel: Warteschlange mit 3 Zuständen gemäss Modell in Abbildung 22:

$$\begin{aligned} \underline{j=0}: \quad \pi_0 v_0 = \pi_1 v_{10} &\Rightarrow \pi_0 \lambda_0 = \pi_1 \mu_1 \Rightarrow \pi_1 = \frac{\lambda_0}{\mu_1} \pi_0 \\ \underline{j=1}: \quad \pi_1 v_1 = \pi_0 v_{01} + \pi_2 v_{21} &\Rightarrow \pi_1 (\lambda_1 + \mu_1) = \pi_0 \lambda_0 + \pi_2 \mu_2 \Rightarrow \pi_2 = \frac{\lambda_0 \lambda_1}{\mu_1 \mu_2} \pi_0 \\ \underline{j=2}: \quad \pi_2 v_2 = \pi_1 v_{12} &\Rightarrow \pi_2 \mu_2 = \pi_1 \lambda_1 \Rightarrow \pi_2 = \frac{\lambda_1}{\mu_2} \pi_1 = \frac{\lambda_0 \lambda_1}{\mu_1 \mu_2} \pi_0 \end{aligned}$$

Im Skript finden sich noch weitere Formeln, deren Herleitung ich allerdings nicht richtig verstanden habe. Komplettheitshalber sollen diese hier dennoch nicht fehlen:

$$\pi_j = \frac{\Pi_j}{\sum_{k=0}^{\infty} \Pi_k} \qquad \Pi_0 = 1 \qquad \Pi_k = \frac{\lambda_0 * \lambda_1 * \dots * \lambda_{k-1}}{\mu_1 * \mu_2 * \dots * \mu_k}$$