

UNIVERZA V LJUBLJANI
FAKULTETA ZA MATEMATIKO IN FIZIKO

Finančna matematika – 1. stopnja

Martin Praček

Skriti markovski modeli v analizi finančnih časovnih vrst

Delo diplomskega seminarja

Mentor:izr. prof. dr. Damjan Škulj

Ljubljana, 2019

KAZALO

Slovar strokovnih izrazov	4
1. Uvod	5
2. Markovski modeli	6
3. Skriti markovski modeli	9
3.1. Zgodovina modela	11
3.2. Predpostavke modela	12
3.3. Priprava in trening modela	14
3.4. Natančna določitev začetnih vrednosti	16
3.5. Trenutno stanje	17
3.6. Generiranje poti v skitem markovskem modelu	18
4. Uporaba	19
4.1. Procesiranje govora	19
4.2. Uporaba v biologiji in biokemiji	19
4.3. Ostali načini uporabe	19
5. Časovne vrste	20
5.1. Dodatne lastnosti modela	21
5.2. Finančna časovna vrsta	22
6. Uporaba skritih markovskih modelov v finančnih časovnih vrstah	23
7. Zaključek	25
Literatura	26

Skriti markovski modeli v analizi finančnih časovnih vrst

POVZETEK

V moji diplomski nalogi sem se ukvarjal z skritimi markovskimi modeli. Gre za vrsto markovskega modela, kjer ne poznamo stanj, v katerih se model nahaja. Ta stanja so torej skrita, od koder tudi izhaja ime skriti markovski modeli. V svojem delu sem opisal osnove teorije skritih markovskih modelov ter nekaj primerov uporabe, obenem pa sem opisal še praktični primer. Posvetil sem se tudi časovnim vrstam in nekaj njihovim posebnim lastnostim.

Hidden Markov Models in Financial Time Series Analysis

ABSTRACT

For my graduate thesis I researched Hidden Markov Models, a type of Markov Models where states of model are not known. This state are therefore hidden, thus naming Hidden Markov Model. My paper discusses the base theory behind hidden markov models and some applications. Also, I included my own example. There is also a part about time series and some of their special properties.

Math. Subj. Class. (2010): 60J05,60J22,91B84, 91B74, 91B70, 91G80,91G70 , 65K05

Ključne besede: skriti markovski modeli, časovne vrste, slučajni proces

Keywords: hidden markov models, time series, stochastic process

Skriti markovski model je statistični markovski model, kjer predpostavljamo, da je modelirani sistem markovski proces s skritimi stanji.

Gaussova mešanica je porazdelitvena funkcija z gostoto, ki jo lahko zapišemo kot tehtano povprečje normalnih gostot;

$$f(x) = \sum_{j=1}^M c_j f_{N,j}(x; \mu_j, \sigma_j^2)$$

. Veljati mora še, da je

$$\sum_{j=1}^M c_j = 1$$

in da je $f_{N,j}(x; \mu_j, \sigma_j^2)$ gostota verjetnosti normalne porazdelitvene funkcije.

Akaikejev informacijski kriterij oziroma AIC je merilo relativne kakovosti modela glede na preostale modele za iste podatke. Naj velja, da imamo podatke s k ocenjenimi parametri in \hat{L} maksimalna vrednost funkcije verjetja. AIC izračunamo kot

$$AIC = 2k - 2\ln(\hat{L})$$

AIC primerja več različnih modelov po kakovosti, kjer je najboljši tisti z najmanjšo vrednostjo.

Baum-Welchov algoritem Je algoritem, s katerim poiščemo neznane parametre skritih markovskih modelov.

Pričakovan primankljaj Naj bo X donos portfelja nekoč v prihodnosti in naj velja da je $0 < \alpha < 1$, kjer je α stopnja zaupanja. Pričakovan primankljaj ES_α je definiran kot

$$ES_\alpha = -\frac{1}{\alpha} \int_0^\alpha VaR_\gamma(x) d\gamma$$

kjer je VaR_γ tvegana vrednost.

1. UVOD

Skriti markovski modeli so modelacijsko orodje, ki nam omogoča zelo široko uporabo. V mojem diplomskem seminarju se najbolj posvečam uporabi v finančni analizi. Z njimi opišemo procese z neznanimi lastnostmi, saj so skriti znotaj podatkov. Prav to nam omogoča dober uvid v finančne vrste.

Skrite markovske modele lahko predstavljamo kot dvojico slučajnih procesov, ki sta med seboj neodvisna. Prvi proces predstavlja stanja, ki jih ne vidimo. Vidimo pa rezultate drugega slučajnega procesa, ki nam vrača signale. Tipičen primer signala je cena delnice.

V drugem poglavju predstavljam osnove teorije za markovskimi modeli, ter razlike med različnimi vrstami le teh, bolj podrobno pa se posvečam markovskih verigam. Razlike med različnimi vrstami markovskih modelov so predstavljene na primeru.

V tretjem poglavju se podrobno posvečam skritim markovskim modelom. Najprej motiviram njihovo uporabo ter predstavil osnovno idejo, ki je bila potrebna za vzpostavitev skritih markovskih modelov. V naslednjem delu tretjega poglavja se posvetim zgodovini skritih markovskih modelov in osebam, ki so osnovalle skrite markovske modele.

Naslednji razdelek je posvečen zahtevam, ki jim moramo ugoditi, da lahko skrite markovske modele modeliramo. Temu sledi tudi razlaga priprave in treninga modela ter določitev začetnih vrednosti.

Naslednja dva dela predstavljata zadnji sklop znotraj poglavja o skritih markovskih modelih. Tu je namreč opisano, kako ugotovimo trenutno stanje ter je orisan algoritem, ki stoji za tem, na koncu pa je opisano še, kako generiramo pot v skritem markovskem modelu.

Naslednje poglavje je posvečeno različnim načinom uporabe, kjer se najbolj podrobno posvetim procesiranju govora in uporabi v biologiji.

Peto poglavje je posvečeno časovnim vrstam ter nekaterim njihovim posebnim lastnostim, POleg tega posebej opišem finančne časovne vrste.

Šesto poglavje je posebej posvečeno uporabi skritih markovskih modelov v finančnih časovnih vrstah. To še podrobneje predstavim z primerom.

2. MARKOVSKI MODELI

Skriti markovski modeli spadajo v bolj splošno skupino modelov, ki jo imenujemo markovski modeli. Gre za modele, kjer se stanja slučajno spreminjajo. Za modele velja, da je njihovo stanje v času t odvisno le od stanja v času $t - 1$. To imenujemo markovska lastnost, po kateri so ti modeli tudi poimenovani.

Definicija 2.1. Naj bo $(\Omega, \mathcal{F}, \mathbb{P}, (\mathcal{F}_s)_{s \geq 0})$ verjetnostni prostor s filtracijo za neko urejeno množico I . Naj velja, da je (S, \mathcal{S}) -merljiv prostor na katerem obstaja merljiv slučajni proces $X = (X_t)_{t \in I}$, ki je prilagojen na filtracijo.

Potem pravimo, da ima slučajni proces markovsko lastnost, če za vsak $A \in \mathcal{S}$ in vsak par $s, t \in I$, kjer velja $s \leq t$, velja

$$\mathbb{P}(X_t \in A \mid \mathcal{F}_s) = \mathbb{P}(X_t \in A \mid X_s).$$

Tu velja, da je Ω množica, \mathcal{F} pa σ -algebra na njej. Ω je množica vseh možnih stanj, v katerih se lahko nahaja neka slučajna spremenljivka. Če tako na primer velja, da naša slučajna spremenljivka lahko zavzame katero koli naravno število, bo potem $\Omega = \mathbb{N}$, \mathcal{F} pa bo enak $\mathcal{F} = 2^{\mathbb{N}}$.

Velja še, da je \mathbb{P} mera na tej množici, kjer velja, da je $\mathbb{P}(\Omega) = 1$, torej je ta mera verjetnostna. Za $(\mathcal{F}_s)_{s \geq 0}$ pravimo, da je filtracija, v posebnem pa je \mathcal{F}_T σ -algebra zgodovine časa T , torej za vsak dogodek do časa T vemo, ali se je dogodek zgodil ali ne.

Zahteva o merljivosti prostora pomeni, da ima množico S podmnožico \mathcal{S} , ki zadošča pogojem za obstoj σ -algebre. Potem je par (S, \mathcal{S}) merljiv prostor.

Slučajni proces je nabor slučajnih spremenljivk in si ga navadno predstavljamo kot zaporedje slučajnih spremenljivk skozi čas. Da je slučajni proces filtracija pomeni, da za vsak n velja, da je X_n za \mathcal{F}_n merljiv.

Enakost torej pravi, da ni razlike med tem, ali bomo pogledali verjetnost $X_t \in A$ pogojno na σ -algebro zgodovine časa s , ali pa pogledamo le, kaj nam pokaže slučajna spremenljivka X_s . To definicijo lahko v primeru markovskih verig še dopolnimo, če velja še, da je $I = \mathbb{N}$. Potem velja

$$\mathbb{P}(X_n = x_n \mid X_{n-1} = x_{n-1}, \dots, X_0 = x_0) = \mathbb{P}(X_n = x_n \mid X_{n-1} = x_{n-1}).$$

Na ta način definiramo markovsko verigo.

Definicija 2.2. Markovska veriga v diskretnem času je zaporedje slučajnih spremenljivk X_1, X_2, X_3, \dots z markovsko lastnostjo, torej je stanje v naslednjem času odvisno le od zadnjega stanja. Pri tem mora veljati, da je

$$P(X_{n+1} = x \mid X_1 = x_1, X_2 = x_2, \dots, X_n = x_n) = P(X_{n+1} = x \mid X_n = x_n),$$

če so pogojne verjetnosti dobro definirane, to je, če velja,

$$P(X_1 = x_1, X_2 = x_2, \dots, X_n = x_n) > 0$$

Gre torej za poseben primer, kjer je imamo slučajni proces v diskretnih časih, ki lahko v vsakem času zavzame le neko diskretno vrednost. V pogoju na levi strani enačaja so sedaj podani vsi dogodki, ki so se zares zgodili, markovska lastnost pa nam pove, da nas zanima zgolj zadnji dogodek.

Primer 2.3. Markovsko verigo si najlažje predstavljamo s pomočjo slučajnih sprehodov. Recimo, da imamo slučajni proces, ki v vsakem času t z verjetnostjo $1/3$

zavzame 1, z verjetnostjo $1/3$ zavzame 0, z verjetnostjo $1/3$ pa zavzame -1 . Torej velja da je $X_t \sim X$ za vsak t in

$$X \sim \begin{pmatrix} -1 & 0 & 1 \\ 1/3 & 1/3 & 1/3 \end{pmatrix}$$

Naj bo sedaj S_n slučajna spremenljivka, ki je definirana kot $S_n = \sum_{t=1}^n X_t$. Recimo, da S_{10} pokaže 3. Zanima nas verjetnost, da bo $S_{11} = 3$. Očitno vidimo, da je ta verjetnost enaka $1/3$, saj velja

$$P(S_{11} = 3) = P(S_{10} + X_{11} = 3 | S_{10} = 3) = P(X_{11} = 0) = P(X = 0) = 1/3$$

. Torej je vrednost naslednjega odvisna le od sedanjega stanja in vrednosti X_{11} , ni pa pomembno, kako smo prišli do $S_{10} = 3$, kar je točno to, kar nam pove markovska lastnost.

◇

V posebnem lahko za markovske verige rečemo še, da so časovno homogene, če velja

$$P(X_n | X_{n-1}) = P(X_1 | X_0).$$

To pomeni, da so vse slučajne spremenljivke v slučajnem procesu enako porazdeljene.

Markovska lastnost nam pove, da je slučajni proces "brez spomina". To pomeni, da pot do stanja ni pomembna, pomembno je le stanje, v katerem se nahajamo. Naravno se nam tudi postavi vprašanje, zakaj je ta lastnost koristna. Izkaže se, da nam omogoča reševanje problemov, ki jih drugače v primernem času ne bi mogli rešiti.

Zaradi svoje široke uporabnosti markovske modele lahko ločimo na:

- prediktivno modeliranje, kjer želimo iz statističnih podatkov izračunati najbolj verjeten dogodek glede nanaše podatke. Ta model se lahko uporablja ne glede na čas v katerem se je dogodek zgodil, saj lahko modeliramo prihodnje dogodke ravno tako kot pretekle dogodke, policija npr. uporablja to vrsto modela za določitev osumljencev za storjeni zločin.
- verjetnostno napoved, kjer ne napovemo le najbolj verjetnega dogodka, ampak določimo verjetnosti vseh dogodkov kot popoln sistem dogodkov. Primer tega so športne stavnice, kjer lahko npr. stavimo za eno izmed ekip stavimo na zmago, remi ali poraz.

Modeliranje markovskih modelov je primer slučajnega programiranja [18]. Gre za sistem, ki nam pomaga pri optimizaciji za modele, ki vključujejo negotovost. To pomeni, da nekateri parametri niso znani z gotovostjo, temveč jih predstavimo z nekim slučajnim objektom, pa naj si bo to slučajna spremenljivka za enoobdobne probleme oziroma slučajni proces za večobdobne probleme.

Glede na to, kako je postavljen markovski model razdelimo, kot kaže tabela 1. Razlike in podobnosti med vsemi modeli so opisane v spodnjem primeru.

Primer 2.4. Primer žabe v ribniku. Zamislimo si, da imamo ribnik z lokvanji, v katerem so se naselile žabe. Vsaka izmed žab je lahko v vsakem trenutku v

TABELA 1. Razdelitev markovskih procesov

	V celoti opazovan	Le delno opazovan
Avtonomen	Markovska veriga	Skriti markovski model
Kontorliran	Markovski odločitveni proces	Delno opazovan proces

vodi, na lokvanju ali pa zunaj vode. To imenujemo stanja markovskega modela S . Poimenujmo vektor stanj

$$S = (Voda, Lokvanj, Zemlja).$$

Na ribnik pogledamo le v nekih diskretnih časih, z enakim razmikom, na primer ob koncu vsake minute. Na ta način naš proces diskretiziramo. Odločimo se za opazovanje le ene izmed žab, za katero nas zanima, kje se bo nahajala v naslednjem trenutku.

Predstavljajmo si, da lahko vsakič, ko pogledamo, našo žabo poljubno osvetlimo z lučjo in tako vplivamo na njeno obnašanje. Naš model je torej kontroliran. Žaba nato poljubno skače naprej. Če poznamo prehodno matriko med stanji, bomo v tem primeru imeli markovski odločitveni proces. To pomeni, da vemo kako nek zunanji vpliv vpliva na spremembe verjetnosti prehoda med stanji. V nasprotnem primeru imamo delno opazovan proces odločanja. To velja ko vplivamo na spremembe stanj, a prehodne matrike ne poznamo.

Če se nikoli ne odločimo, da bi žabo osvetlili, pa imamo avtonomen model. Kot pri kontroliranem lahko tu ločimo, ali je proces v celoti opazen ali so neka stanja skrita. Če stanja niso skrita, imamo markovsko verigo, ki je torej le poseben primer markovskih odločitvenih procesov, kjer žabe nikoli ne prestavimo. Pri markovskih verigah poznamo tudi prehodno matriko A , katere elementi a_{ij} predstavljajo verjetnosti prehoda iz stanja i v stanje j . Preprost primer markovskih verig predstavljajo slučajni sprehodi.

Navadno predpostavimo, da se verjetnosti prehoda iz enega stanja v drugega ne spreminjajo. Vsota po vsaki vrstici je vedno enaka ena, to je

$$\sum_{j=1}^n a_{ij} = 1$$

saj predstavlja verjetnosti prehoda iz enega stanja v drugo. Elementi Tako je a_{ij} verjetnost prehoda iz stanja i v stanje j . Velja še, da je n število stanj in je v našem primeru 3, očitno pa velja tudi, da je matrika velikost $n \times n$ in da so vsi elementi a_{ij} iz intervala $[0, 1]$.

V tem primeru lahko torej, v vsakem trenutku predvidimo, kam bo skočila žaba v naslednjem trenutku. Če stanj in prehodov med njimi ne poznamo imamo skrite markovske modele.

◇

3. SKRITI MARKOVSKI MODELI

Skriti markovski model je statistični markovski model, kjer predpostavljamo, da je modelirani sistem markovski proces s skritimi stanji. Gre torej za tip modela, kjer lahko vidimo signal, ne moremo pa ugotoviti, kakšna je bila funkcija, ki nam ga je dala.

Primer 3.1. Za žabo iz primera 2.4 sedaj pogledjmo kaj se spremeni v primeru skritih markovskih modelov.

Za le te velja, da vseh parametrov ne poznamo, lahko pa jih določimo iz podatkov, ki jih imamo za naš problem. Tako bi našo žabo dovolj dolgo opazovali, in si zapisovali na primer njeno temperaturo v vsakem trenutku opazovanja. Tako bi dobili vektor opazovanj O , ki bi vseboval temperaturo žabe v vsakem trenutku. V pomočjo skritih markovskih modelov bi nato poizkušali določiti v kakšnem stanju, torej na kakšni poziciji se je nahajala v vsakem trenutku. Določitev zaporedja stanj je se razlikuje med primeri. V primeru žabe bi tako lahko pogledali spremembo temperature ob spremembi stanja, ter glede na to spremembo poizkušali določiti, kje se žabica nahaja.

Vektor O je naš osnovni podatek, iz katerega nato izpeljemo celoten model, torej poiščemo zaporedje skritih stanj in nato verjetnosti prehodov med njimi. Za Iz tega bi najprej pogledali, kakšna so možna stanja sistema. Ko bi določili stanja ter njihovo zaporedje, iz tega lahko določimo parametre modela. Kako to naredimo, si bomo pogledali v nadaljevanju, na primeru prehodne matrike pa pogledjmo, kaj le ta pomeni.

Recimo, da prehodna matrika enaka:

$$\begin{bmatrix} 0.5 & 0.3 & 0.2 \\ 0.1 & 0.5 & 0.4 \\ 0.4 & 0.2 & 0.4 \end{bmatrix}$$

V našem primeru bi to pomenilo, da je verjetnost, da žaba ostane v vodi 0.5, da iz vode skoči na lokvanj 0.3, verjetnost, da pa skoči iz ribnika pa 0.2. Iz te matrike lahko torej preberemo verjetnosti, ki nam bodo povedale verjetnosti prehodov, če velja da na naš opazovan sistem, torej žabo, ne vplivamo.

Kot smo prej omenili, pa se ta matrika spremeni, če žabo osvetlimo. Tako lahko dobimo matriko, ki govori o verjetnostih prehodov, če na sistem vplivamo.

$$\begin{bmatrix} 0.9 & 0.05 & 0.05 \\ 0.4 & 0.5 & 0.1 \\ 1 & 0 & 0 \end{bmatrix}$$

Le ta nam pove, da je verjetnost da ostane v vodi enaka 0.9, medtem ko je verjetnost, da od zunaj skoči v vodo enaka 1. Tu je matrika torej drugačna, kot je bila za prvi primer. Na vsakem koraku, ko žabo osvetlimo moramo torej upoštevati to drugo matriko.

Kontroliran sistem torej pomeni, da lahko z nekim znanim dejanjem vplivamo na prehodne verjetnosti.

◇

Skrit markovski model je končno učljiv slučajen avtomat. Lahko ga opišemo kot dvojni slučajni proces. Prvi proces ima končno število stanj. Med temi prehode

opazujemo prek prehodnih verjetnosti. Pri drugem slučajnem procesu lahko v vsakem stanju opazujemo signal. Signali so odvisni od stanja prvega procesa. Drugi proces namreč vrne signal pogojno na zavzeto vrednost prvega procesa.

Ker te signale opazujemo brez tega, da bi vedeli, v kakšnem stanju so se zgodili, imenujemo ta stanja skrita.

Skriti markovski modeli uporabljajo idejo bayesijske statistike. Tu uporabljamo Bayesov izrek ??, ki pravi, da je verjetnost dogodka A , v primeru da B velja enaka:

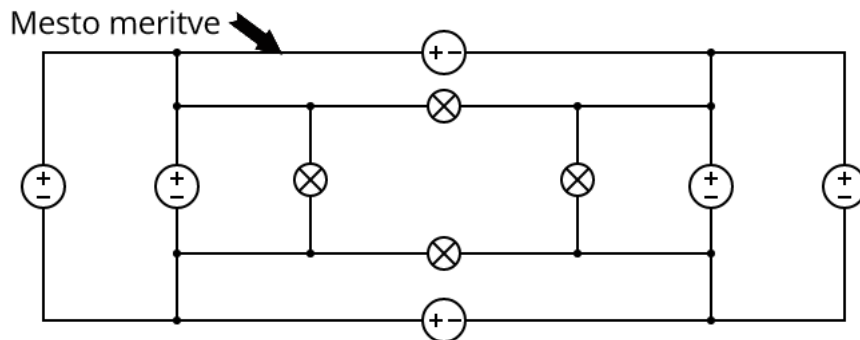
$$P(A|B) = \frac{P(B|A)P(A)}{P(B)},$$

kjer velja, da dogodek A predstavlja našo hipotezo, B pa znan dogodek.

Ideja za delovanje skritih markovskih modelov izhaja iz dela Andrewa Viterbija na področju komunikacije. V komunikacijskih storitvah šumi namreč predstavljajo veliko težavo, zato se je ameriško-italijanski elektroinžiner domislil algoritma, s katerimi bi iz seta podatkov izločili šume.

Primer 3.2. To si lahko predstavljamo kot da bi si na primer želeli na fiksnem mestu v električnem tokokrogu določiti naboj $Q(t)$ glede na naboj ob času s , kjer $s < t$. Zaradi napake pri merjenju $Q(s)$ ne moremo točno določiti, saj zaznamo neko zmoteno različico $\widehat{Q}(s)$.

Želimo si torej način, ki bi čim bolj učinkovito izločil te napake. V našem primeru



SLIKA 1. Skica primera električnega tokokroga z mestom meritve

imamo 6 električnih virov in 4 žarnice. Po 30 sekundah, ko zaženemo tokokrog, izmerimo naboj na mestu meritve. Zanima nas, ali je naboj po minuti še vedno enak.

Neznana stanja so poleg dejanskega naboja, ki ga želimo izmeriti, še šumi, ki se prenašajo po omrežju. Prav ti šumi namreč pripeljejo do napake pri meritvi.

◇

Skrite markovske modele navadno predstavimo na diskretnih množicah in v diskretnem času, a lahko definicijo tako razširimo, da velja tudi za zvezne množice

ali zvezen čas. V tem primeru dobimo triplasten slučajni proces, katerega največja težava je velika računska zahtevnost, poleg tega pa nam lahko veliko število parametrov prinese nestabilnost. Vendar se z zveznimi primeri v svojimi diplomski nalogi ne bom ukvarjal.

Prav tako je možno s skritimi markovskimi modeli predstaviti procese, odvisne od več slučajnih spremenljivk. Vendar tak model s seboj prinese več težav. Prve se pokaže v računski zahtevnosti, saj moramo v vsakem koraku določiti inverz in determinanto več matrik, kjer lahko časovno zahtevnost povečuje tudi njihova velikost. Drugi problem se pokaže v samem računanju, ki je podrobneje razloženo v 3.3. Izkaže se, da je v primeru prevelikega števila slučajnih spremenljivk včasih nemogoče izvesti celo prvi korak iteracije. Modeliranje skritih markovskih modelov z več spremenljivkami je tako v teoriji možno, a ni nujno dobro v praksi.

3.1. Zgodovina modela. Ideja skritih markovskih modelov se v svoji prvi fazi začne z ruskim matematikom Andrejem Andrejevičem Markovom, ki je za časa svojega življenja med 1865 – 1922 razvil idejo markovskega procesa in verige. Prve teoretične rezultate markovskih verig je svetu predstavil 1906, kar najdemo v [11], 1913 pa je izračunal zaporedje črk ruščine [12]. Slednji izračun je opravil na besedilu Jevgenij Onjegin ruskega pesnika Aleksandra Sergejeviča Puškina. Markov je želel s tem dokazati veljavnost šibkega zakona velikih števil zaradi spora z drugim ruskim matematikom Pavlom Nekrasom, a je na ta način odprl novo vprašanje v matematiki.

Skriti markovski modeli zahtevajo veliko numeričnega računanja, in zato ne preseneča, da se je nadaljni razvoj začel šele z širšo uporabo računalnikov. Po pomembnem delu, ki so ga opravili von Neumann, Turing in Conrad Zuse so se znanstveniki začeli pospešeno ukvarjati z implementacijo primernih algoritmov. Velik korak k temu je pripomogel Claude Shannon s svojim delom Matematična teorija komunikacije [10].

Za implementacijo skritih markovskih modelov je bilo v začetku potrebnih kar nekaj razvojev algoritmov. Prvi algoritem je bil algoritem maksimizacije pričakovane vrednosti (Expectation-maximization), ali, kot ga med drugim poznamo, EM-algoritem. Tega so uporabljali med drugim že Laplace, Gauss in drugi, vendar je do dokončnega poimenovanja prišlo šele leta 1977. Dempster, Laird in Rubin so algoritem poimenovali in razložili splošno teorijo, ki deluje v ozadju procesa v [13]. Andrew Viterbi je drugi pomemben del skritih markovskih modelov implementiral leta 1967 kot algoritem za dekodiranje konvolucij čez šume v komunikaciji v [14]. Pri tem algoritmu deluje princip dinamičnega programiranja, ki najde najbolj verjetno zaporedje stanj, glede na dano opazovano zaporedje dogodkov. Viterbijev algoritem je podrobneje opisan v algoritmu 1.

Celotnih modelov pa prav gotovo ne bi bilo brez še enega Rusa, in sicer Ruslana Leontijeviča Stratonoviča, ki je prvi opisal rekurzijo naprej - nazaj leta 1960 v [16]. To je delal na primeru optimalnega nelinearnega problema filtracije.

Za glavnega avtorja pa velja Leonard E. Baum. Skupaj z Lloyd R. Welch sta okrog leta 1970 razvila Baum-Welchov algoritem v [15], kjer gre za poseben primer posplošenega EM-algoritma. Prav Baum-Welchov algoritem je tisti korak, pri katerem velja, da je bila dokončno izpeljana teorija za skritimi markovskimi modeli. Algoritem uporablja EM-algoritem da najde cenilko po metodi največjega verjetja glede na dano zaporedje podatkov. Delovanje algoritma je podrobno razloženo v 3.3, njegove zahteve pa v 3.2.

Uporabe skritih markovskih modelov na področju financ in ekonomije pa ne bilo brez dela ameriškega ekonometrista Jamesa Hamiltona, ki je predlagal, da bi model, katerega stanj in prehodov med njimi ne poznamo, opišemo z markovskim modelom v [17].

3.2. Predpostavke modela. Kot vsak matematični model, ima tudi skriti markovski model svoje predpostavke.

- (1) Prva predpostavka je, da lahko rezultate, ki jih model producira, opazujemo v ekvidistančnih časih, torej je razlika med dvema poljubnima zaporednima časoma opazovanja $t - 1$ in t vedno enaka, na primer d . Rezultate imenujemo signali ali opazovanja.
- (2) V vsakem izmed časov t je sistem lahko v enem izmed N stanj. Ta stanja so S_1, S_2, \dots, S_N in so v času t neznana, vemo le, kakšni so rezultati našega procesa v tem času. Zato potrebujemo dodatni slučajni proces $Q = (Q_t)_{t=1,2,\dots}$, ki nam pove, kakšno je opazovanje v času t . Tako velja, da je signal O_t dan s stanjem slučajnega procesa Q . V moji diplomski nalogi bom to podrobneje opisal v odvisnosti od gostote verjetnostne porazdelitve Gaussove mešanice.
- (3) Začetna porazdelitev po stanjih je označena z Π , in vsota njenih elementov je enaka 1. Dolžina tega vektorja je N , torej je enaka številu možnih stanj. Velja, da je Π_i verjetnost, da se v začetku nahajamo v stanju i .
- (4) V vsakem času velja, da bodisi sistem spremeni svoje stanje, bodisi ostane v istem. Verjetnost prehoda v vsako stanje je določena s prehodnimi verjetnostmi, podanimi v matriki A_t , kjer verjetnost prehoda iz stanja i v času t v stanje j v času $t + 1$ simbolizira element a_{ij}^t . Vsota vsake vrstice vsake matrike A_t je enaka 1, saj bo v naslednjem času očitno nekam prešla. Matrika A_t je velikosti $N \times N$, kjer je N število stanj. To velja, ker nam pove verjetnosti prehodov med poljubnimi stanji.
- (5) Ker govorimo o markovskem modelu, bo stanje v času $t + 1$ odvisno le od stanja v času t , ne pa od celotne zgodovine.

Za konstrukcijo modela te predpostavke zadoščajo, a upoštevamo še nekaj dodatnih, ki nam model še precej olajšajo.

- (1) Predpostavimo lahko, da so vse prehodne matrike A^t enake, torej neodvisne od časa t , kar označuje časovno homogenost. Enolično prehodno matriko torej lahko označimo z A .
- (2) Signal v času t je odvisen le od stanja modela v tem času; torej so slučajne spremenljivke v času t odvisne le od tega.

Naši signali so torej odvisni od stanja S . Ta stanja nam vrnejo rezultat v odvisnosti od verjetnostnih porazdelitev podanih z različnimi parametri. Ločimo lahko signale, ki so porazdeljeni kot zvezne slučajne spremenljivke, ter kot diskretne slučajne spremenljivke.

V zveznem primeru bodo le-te predstavljene kot mešanice normalnih porazdelitev in jih imenujemo tudi Gaussove mešanice.

Gaussovo mešanico definiramo kot tehtano povprečje normalnih porazdelitvenih

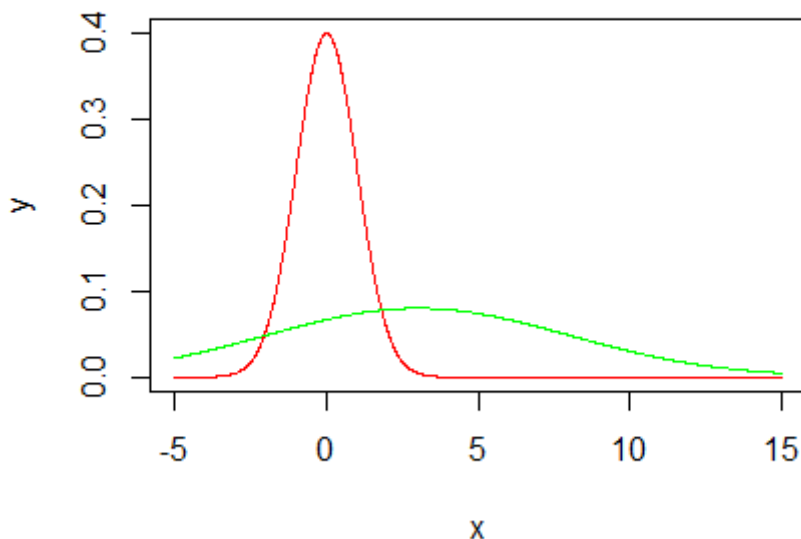
funkcij, glej slovar strokovnih izrazov. Vsako stanje i iz nabora vseh stanj je torej podano z gostoto verjetnosti,

$$b_i(x) = \sum_{j=1}^M c_{ij} N(x; \mu_{ij}, \sigma_{ij}^2)$$

Pri nekem stanju i je b_i signal, ki ga prejmemo. V primeru finančnih časovnih vrst bi torej to bila cena finančnega instrumenta. Cena je realna vrednost, ki jo vrne slučajna spremenljivka, odvisna od stanja.

Primer 3.3. Gaussovo mešanico lahko vidimo na spodnjih dveh slikah.

Na prvi sliki imamo dve gostoti normalne porazdelitvene funkcije. Prva je stan-



SLIKA 2. Ločeni normalni porazdelitvi

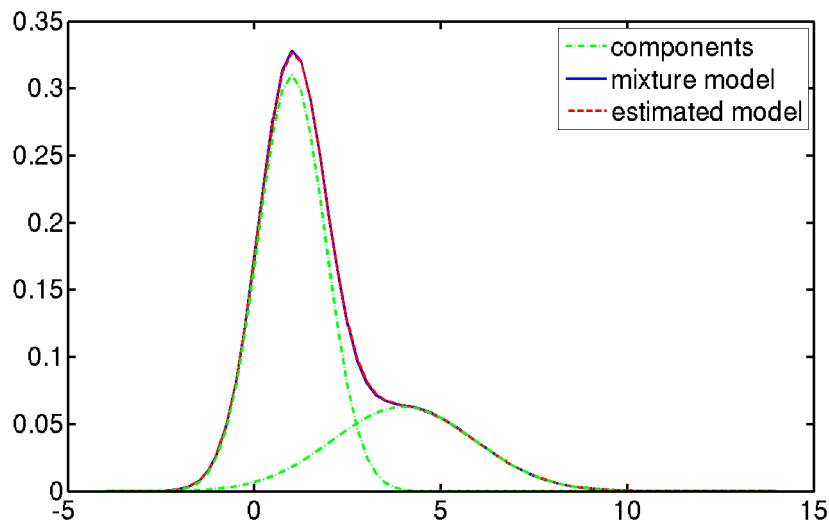
darno normalno porazdeljena, torej $N(0, 1)$, druga pa je porazdeljena $N(3, 5)$.

Na drugi sliki pa vidimo kako z Gaussovo mešanico ocenimo skupno porazdelitev. Skupna porazdelitev je torej v našem primeru enaka

$$b(x) = c_1 N(x; 0, 1) + c_2 N(x, 3, 5),$$

kjer za c_1 in c_2 velja, da je $c_1 \geq 0$ in $c_2 \geq 0$ ter $c_1 + c_2 = 1$. Tako dobimo vsoto dveh normalnih porazdelitev in vsoto njunih porazdelitev. To je tudi konveksna kombinacija normalnih porazdelitev. \diamond

Gaussove mešanice so primerne, ker lahko zelo dobro aproksimirajo vsako končno zvezno porazdelitev, poleg tega pa se znebimo tveganja, ki nam ga v tem modelu



SLIKA 3. Ocena z Gaussovo mešanico

predstavlja normalna porazdelitev. To tveganje predstavlja dejstvo, da je normalna porazdelitev simetrična; to pa pogosto ne drži za procese v realnem svetu, na primer za donose v financah. Za te večkrat drži, da so asimetrične v levo, torej da je mediana večja od povprečja.

Tako potrebujemo za sestavo modela še:

- število mešanic M
- matriko C , ki predstavlja koeficiente c_{ij} , ki so faktorji v Gaussovi mešanici. Matrika je velikosti $N \times M$ in element (i, j) pove relativno težo, ki jo ima mešanica j v aproksimaciji S_i .
- matrika Γ , kjer μ_{ij} predstavlja pričakovano vrednost b_j v stanju i , ter
- matriko Σ , kjer σ_{ij} predstavlja varianco b_j v stanju i .

Vse te matrike so velikosti $N \times M$ in z njimi opišemo Gaussove mešanice.

Če so slučajne spremenljivke, ki opisujejo signale diskretne, je opis stanj manj kompleksen. V tem primeru za opis diskretne spremenljivke potrebujemo le eno matriko, katere elementi opišejo vrednosti, ki jih dobimo. To matriko imenujemo izhodna matrika. Ta matrika je velikosti $N \times N$, kjer je N število stanj. Ta matrika vsebuje vrednosti, ki jih vrne slučajni proces kot signale.

3.3. Priprava in trening modela. Preden začnemo z določanjem parametrov našega modela, moramo najprej dobro preučiti naš problem, saj je to znanje ključno za dobro reševanje problema. Kot prvo stvar moramo najprej izbrati set podatkov, na katerem bo potekal tako imenovan trening sistema. Ti podatki morajo biti zbrani na na primerni enoti; na ceno delnic vrednost le te pred 10 leti ne bo vplivala toliko kot cena dan pred meritvijo.

Podatke, ki smo jih zbrali moramo nato urediti; določimo časovne trenutke z enakim razponom, kot ga želimo imeti v našem modelu, in v teh trenutkih $t \in (1, \dots, T)$ določimo opazovanja O .

Ko imamo podatke zbrane, moramo najprej določiti število stanj N ter število

mešanic M . Tu gre tudi za ključna problema priprave skritega markovskega modela. V praksi število stanj včasih določimo v skladu s potrebno aplikacijo, kot na primer v 4.2, oziroma z vizualnim ogledom točk grafa iz zgodovinskih podatkov.

Včasih pa lahko to določimo z uporabo Akaikejevega informacijskega kriterija, ki primerja več različnih modelov in izbere najboljšega. AIC ocenjuje relativno kakovost modela. Podobno deluje tudi BIC, to je Bayesov informacijski kriterij.

Za pravilno določitev števila mešanic M pa navadno izberemo razvrščanje v skupine (k -mean clustering). S to metodo želimo n opazovanj razvrstiti v k skupin, kjer upoštevamo, da želimo minimizirati skupno varianco in število skupin.

Ostale parametre navadno označimo z $\lambda = (\Pi, A, C, \Gamma, \Sigma)$. Cilj treninga modela je, da parametre nastavimo tako, da bo verjetnost $P(O|\lambda)$, da so bila vsa opazovanja rezultat modela, največja. Pri tem mora veljati, da sta število mešanic M ter število stanj N že znana, $P(O|\lambda)$ pa je funkcija verjetja.

Naravno se pojavi vprašanje, izbora primernih parametrov. Izkaže se, da je učinkovit sistem *naprej – nazaj*. Določimo spremenljivki *nazaj* $\alpha_t(i)$, ki predstavlja verjetnost, da so se zgodila opazovanja od 0 do t in stanje i , ter *naprej* $\beta_t(i)$, ki predstavlja verjetnost, da se bodo zgodila opazovanja od t do T pri stanju i . Posebej pa se moramo posvetiti le eni izmed njiju, saj gre pri $\beta_t(i)$ le za alternativno metodo za $\alpha_t(i)$. $\alpha_t(i)$ definiramo kot:

$$\alpha_1(i) = \pi_i b_i(O_1) = \pi_i b_i = \sum_{k=1}^M c_{ik} N(O_1; \mu_j, \sigma_j^2)$$

Naslednje $\alpha_{t+1}(j)$ lahko nato induktivno izračunamo kot

$$\alpha_{t+1}(j) = b_j(O_{t+1}) \sum_{i=1}^N \alpha_t(i) a_{ij}$$

Iz tega sledi, da je $P(O|\lambda) = \sum_{i=1}^N \alpha_T(i)$. Na podoben način izračunamo tudi $\beta_t(i)$. β je inicializirana z

$$\beta_T(i) = 1, \forall i \in 1, \dots, N$$

indukcija pa poteka prek

$$\beta_t(i) = \sum_{j=1}^N \beta_{t+1}(j) a_{ij} b_j(O_{t+1})$$

$\beta_t(i)$ sicer ni nujen za izračun $P(O|\lambda)$, a ga nujno potrebujemo za trening našega modela.

Na ta način smo opisali Baum-Welchov algoritem, ki je osnova ocenjevanja skritih markovskih modelov.

Skozi celotno točko 3.3 smo se obnašali, kot da lahko vse parametre iz λ kar izračunamo. Vendar to ne drži v popolnosti. Res, ko je enkrat *naprej – nazaj* vzpostavljen, se nam dozdeva, da gre le še za numerično operacijo. Vendar težave nastopijo preden naš postopek začne z delom. Problem nastopi ko moramo vnaprej oceniti začetne parametre λ .

Ne poznamo analitičnega načina, kako bi lahko ta problem rešili, vendar lahko najdemo tak λ , da lahko našo verjetnost $P(O|\lambda)$ lokalno maksimiziramo, ne moremo pa zagotoviti globalnega maksimuma. Prav tako se lahko zgodi, da predobro opišemo nek set podatkov, to pomeni, da je $P(O|\lambda_{dejanski}) \leq P(O|\lambda_{zadnji})$. Tako se lahko zgodi, da svoji oceni preveč zaupamo. To imenujemo tudi preprileganje ali overfit.

Dejstvo, da ne vemo če smo našli globalni maksimum, ali le lokalni pomeni, da lahko že manjša perturbacija parametrov pomeni spremembo lokalnega maksimuma in tako spremembo končnih rezultatov. Občutljivost sistema je odvisna števila parametrov.

Pri tem moramo vsak parameter λ najprej pogledati posebej. Izkaže se, da začetni vrednosti za prehodno matriko A in začetni vektor Π nista pomembni, če nista ničelna, kar pa se ne more zgoditi. Za Π tako lahko vzamemo vektor, kjer so vse vrednosti enake $1/N$, kjer je N število stanj.

Več problemov nam povzročajo tri postavke z zvezno porazdelitvijo, to je C , Σ in Γ . Dobra začetna ocena le teh je nujna za kakovost modela. Tudi tu se izkaže, da lahko, podobno kot pri oceni števila mešanic M obrnemo na razvrščanjem v skupine. Ta postopek nam sicer ne da globalnega maksimuma, ki bi ga mogoče lahko dobili na drugačen način, a se v praksi izkaže za dobrega. Za C to pomeni, da bo element $c_{ij} = 1/k$ za vsak par (i, j) , kjer je k število šopov povprečij. Pričakovane vrednosti in variance nato pridobimo iz vrednosti šopov povprečij.

Z znanimi začetnimi vrednostmi se lahko spustimo v maksimiziranje $P(O|\lambda)$. Osnovati moramo zaporedje $\lambda = \lambda_t, t \in 0, \dots$, da bo veljalo

$$P(O|\lambda_{i+1}) \geq P(O|\lambda_i)$$

Za to zaporedje velja, da konvergira k lokalnemu maksimumu.

3.4. Natančna določitev začetnih vrednosti. Zanima nas torej, kako bomo osnovali zaporedje λ , ki ga potrebujemo za maksimizacijo $P(O|\lambda)$. Izkaže se, da za maksimizacijo le tega potrebujemo še nekaj dodatnih formul, ki so rezultat Baum-Welchovega algoritma. Z njimi definiramo večfazni iterativni proces, pri katerem popravljamo vrednosti parametrov do konvergence.

Kot prvo potrebujemo $\xi_t(i, j)$, ki nam pove pogojno verjetnost, da smo v času t v stanju i in v času $t+1$ v stanju j , glede na vektor opazovanj O in nabor parametrov λ .

$$\xi_t(i, j) = \frac{\alpha_t(i) a_{ij} b_j(O_t) \beta_{t+1}(j)}{P(O|\lambda)}.$$

Vsota $\sum_{t=1}^N \xi_t(i, j)$ nam pove pričakovano število prehodov iz stanja i v stanje j . Druga je $\gamma_t(i)$, ki nam pove verjetnost, da smo v času t v stanju i .

$$\gamma_t(i) = \frac{\alpha_t(i) \beta_t(i)}{P(O|\lambda)}$$

Vsota $\sum_{t=1}^N \gamma_t(i)$ nam pove pričakovano število prehodov iz stanja i .

Zadnja, tretja pa je $\gamma_t(j, k)$, ki nam pove verjetnost, da smo v času t v stanju j , in k ta mešanica določa O_t .

$$\gamma_t(j, k) = \gamma_t(j) \frac{c_{jk} N(x; \mu_{jk}, \sigma_{jk}^2)}{\sum_{m=1}^M c_{jm} N(x; \mu_{jm}, \sigma_{jm}^2)}$$

Iz teh podatkov lahko nato popravimo izračun parametrov:

$$\begin{aligned} \overline{\Pi}_i &= \gamma_1(i) \\ \overline{a_{ij}} &= \frac{\sum_{t=1}^{T-1} \xi_t}{\sum_{t=1}^{T-1} \gamma_t(i)} \end{aligned}$$

$$\overline{c_{jk}} = \frac{\sum_{t=1}^T \gamma_t(j, k)}{\sum_{t=1}^T \sum_{m=1}^M \gamma_t(j, m)}$$

$$\overline{\mu_{jk}} = \frac{\sum_{t=1}^T \gamma_t(j, k) O_t}{\sum_{t=1}^T \gamma_t(j, k)}$$

Ko je to določeno, nam zaporedje λ , določenih s temi parametri da lokalni maksimum.

3.5. Trenutno stanje. Zadnja stvar, ki jo moramo narediti, preden začnemo z določanjem prihodnjih stanj je določitev trenutnega stanja gospodarstva, torej stanja v zadnjem času, na katerem je naš model treniral. Za določitev le tega uporabimo tako imenovani Viterbijev algoritem. Le ta je oblikovan tako, da nam vrne zaporedje Q , ki maksimizira $P(Q|O, \lambda)$.

Algorithm 1: Viterbijev algoritem

Input:

- Prostor opazovanj $O = \{o_1, o_2, \dots, o_N\}$
- Prostor stanj $S = \{s_1, s_2, \dots, s_K\}$
- Nabor začetnih verjetnosti $\Pi = (\pi_1, \pi_2, \dots, \pi_K)$,
kjer velja da je $\pi_i == s_i$
- Zaporedje opazovanj $Y = (y_1, y_2, \dots, y_T)$,
tako da je $y_t == i$, če je opazovanje v času t o_i
- Prehodna matrika A velikosti $K \times K$,
kjer stanje A_{ij} pove verjetnost prehoda iz stanja s_i v stanje s_j
- Izhodno matriko B velikosti $K \times N$,
kjer stanje B_{ij} pove verjetnost, da opazimo o_j iz stanja s_i

```

for vsako stanje  $j \in \{1, 2, \dots, K\}$  do
  |  $T_1[j, 1] \leftarrow \pi B_{jy_1}$ 
  |  $T_2[j, 1] \leftarrow 0$ 
end
for vsako opazovanje  $i = 2, 3, \dots, T$  do
  | for  $j \in \{1, 2, \dots, K\}$  do
  | |  $T_1[j, i] \leftarrow \max_k (T_1[k, i-1] \cdot A_{kj} \cdot B_{jy_i})$ 
  | |  $T_2[j, i] \leftarrow \arg \max_k (T_1[k, i-1] \cdot A_{kj} \cdot B_{jy_i})$ 
  | end
end
 $z_T \leftarrow \arg \max_k (T_1[k, T])$ 
 $x_T \leftarrow s_{z_T}$ 
for  $i \leftarrow T, T-1, \dots, 2$  do
  |  $z_{i-1} \leftarrow T_2[z_i, i]$ 
  |  $x_{i-1} \leftarrow s_{z_{i-1}}$ 
end
Result: Najbolj verjetno zaporedje opazovanj  $X = (x_1, x_2, \dots, x_N)$ 

```

Za delo s tem algoritmom moramo definirati $\delta_t(i)$, ki za vsako stanje i vrne največjo verjetnost vzdolž poti v času t . Prek $\delta_t(i)$ nato induktivno izvedemo algoritem.

Viterbijev algoritem nam vrne p^* , ki je največja verjetnost in q_T^* , ki nam pove

stanje v času T , ki nam to verjetnost vrne.

3.6. Generiranje poti v skritem markovskem modelu. V celoti lahko sedaj predstavimo delovanje skritega markovskega modela na primeru generiranja ene poti. V tem primeru je pot na primer cenovni proces, ki ga bo glede na naš model opravil finančni instrument.

Glede na naše podatke, ki jih označimo kot $O = (O_1, \dots, O_T)$, ocenimo parametre našega modela $\lambda = (\Pi, A, C, \Gamma, \Sigma)$. Začetne vrednosti elementov Π, A in C predpostavimo kot enake, za Γ in Σ pa ocenimo prek razvrščanja v skupine. Te parametre rekurzivno izboljšujemo, dokler nismo zadovoljni z določeno $P(O|\lambda)$. Parametre, ki smo jih dobili na ta način lahko za isti set podatkov uporabimo poljubno mnogokrat. Določiti moramo še trenutno stanje našega modela. V tem trenutku določimo še čas, do katerega želimo oceniti naš proces, naj bo to recimo T_T .

Naslednji korak si lahko predstavljamo kot zanko. Če predpostavimo, da naš parameter t teče od časa T do časa T_T , v vsakem koraku glede na stanje, v katerem se nahajmo, izberemo stanje, v katero se bomo premaknili v naslednjem koraku. V vsakem koraku še shranimo stanje, v katerem smo bili. Na ta način dobimo pot, ki jo generira skriti markovski model.

Ta postopek lahko ponovimo večkrat, kar se pomaga pri Monte Carlo simulaciji.

Algorithm 2: Monte Carlo simulacija skritih markovskih modelov

Input: Podatki $O = (O_1, \dots, O_T)$
 Ocenimo parametre λ
while *Dokler* $P(O|\lambda)$ *ni dovolj velik* **do**
 | Ponovno izračunaj $P(O|\lambda)$ z novimi parametri λ
end
 Ocenimo stanje v času T z Viterbijevim algoritmom
for $s = 1, \dots, S$, *kjer je* S *število stanj* **do**
 Dobimo vzorec iz izhodne porazdelitve stanja i , kjer je $t = 1$
 while $t \leq TP$, *kjer je* TP *zadnji čas, kjer nas vrednost zanima* **do**
 | $t = t + 1$
 | Izberi stanje za čas t v skladu z prehodno matriko A
 | Generiramo vzorec iz izhodne porazdelitve tega stanja
 end
end
Result: Zaporedje stanj

4. UPORABA

Skriti markovski modeli so zelo široko uporabno orodje za modeliranje. Načini uporabe se zelo razlikujejo, od uporabe v financah do določevanja genoma. Ekonomskim, torej tistim, ki jih delujejo kot napovedovalci cen vrednostnih papirjev v prihodnosti, se bom najgloblje posvetil v naslednjem delu, zato sedaj raje pogledjmo ostale načine uporabe.

4.1. Procesiranje govora. Prva ideja Leonarda E. Bauma za njegov algoritem je bila prav uporaba v procesiranju govora. Ena najbolj široko uporabljenih načinov uporabe pa je procesiranje govora za posamične glasovne enote. Gre za sistem, kjer želimo prepoznati posamične izgovorjene besede, ne moremo pa ga uporabiti za splošen govor, saj algoritem ni implementiran za razumevanje tekočega govora, le za posamezne enote. Ideja tega algoritma je razviti najboljše možne aproksimacijske algoritme, da lahko skriti markovski model filtrira naključne šume, zvoke na najboljši možen način.

Da bomo procesiranje lahko zmodelirali, moramo najprej določiti slovar glasov, iz katerega vemo, da bo glas prišel. Tu uporabljam besedo glas, ker ni nujno, da bo prepoznan glas dejansko beseda; eden izmed glasov bi lahko bil tudi zgolj črka A.

4.2. Uporaba v biologiji in biokemiji. Gre za eno izmed prvih uporab skritih markovskih modelov. Ker se je uporaba na tem področju začela precej zgodaj tudi ne preseneča, da je uporaba precej široka. Uporaba skritih markovskih modelov se je v razširila zaradi povečanja količine podatkov, ki je znanstvenikom na voljo. Ker so v zadnjem času določili več zaporedij genoma, je tako količino podatkov težko obdelati brez uporabe naprednih računskih metod.

Kot že rečeno, je uporaba skritih markovskih modelov v biologiji zelo široka. Uporablja se jih kot že rečeno za obdelave genoma, za določanje genov, za modeliranje procesov v celični membrani. Splošno prepričanje je, da se bo uporaba skritih markovskih modelov v biologiji in ostalih povezanih znanostih še razširila.

Pomemben primer uporabe v primeru določanja integralnih membranskih proteinov. To je protein, ki je stalno pripet v biološko membrano, saj predstavljajo 20 – 30% proteinov v popolnoma sekvenciranih genomih. Tu ločimo dve vrsti proteinske strukture, α in β . Določanje α danes ni več problem, se pa zato pojavljajo precej večji problemi pri β membranskih proteinih. Za ta primer so avtorji [7] razvili poseben primer skritega markovskega modela.

Pomembna je tudi uporaba za lociranje genov, torej iskanje dela genomske deoksiribonukleinske kisline (DNK), ki kodira gene.

4.3. Ostali načini uporabe. Skrite markovske modele se uporablja še na precej ostalih primerih. Skrite markovske modele se tako uporablja za

- prevajanje govora oziroma pisave iz enega jezika v drug jezik,
- kriptanaliza, torej omogočanje vdora v zakodirane sisteme,
- prepoznavanje lastnoročne pisave,
- prepoznavanje dejanj,
- napovedovanje prevoza.

Skriti markovski modeli so torej široko uporabno modelacijsko orodje.

5. ČASOVNE VRSTE

Definicija 5.1. Časovna vrsta je množica opazovanj x_t , vsako opazovano ob časih t znotraj nekega časovnega intervala.

Definicija 5.2. Model časovne vrste za opazovane podatke x_t je slučajni proces X_t , kjer velja, da so x_t realizacije tega slučajnega procesa v časih t .

Vrednosti delnic po dnevih, število uporabnikov spletne strani po urah, prebivalstvo neke države po letih, ... To je le nekaj izmed mnogih primerov uporabe časovnih vrst. Časovne vrste nam dajo lepšo idejo o trendu podatkov, ki jih želimo analizirati. Enostavno lahko vidimo, ali je npr. cena jagod odvisna od letnega časa ter ali se cena jabolk skozi leta spreminja.

Časovna analiza je lahko analitična, kjer preučujemo podatke iz preteklosti, ter prediktivna, kjer želimo napovedati neke trende. Analize lahko poteka na več načinov, od linearnih in nelinearnih modelov do modelov z eno spremenljivko in modelov z več spremenljivkami. Prav zaradi njihove široke uporabnosti se je razvilo zelo veliko modelov za njihovo analizo, začnši leta 1970 z delom Boxa in Jenkinsa na področju avtoregresivnega integriranega drsečega povprečja, danes znanega pod kratico ARIMA.

Časovne vrste so po definiciji odvisne od časa. Prav zato je pomembno, da se opazovanj med seboj medčasovno ne menja. Opazovanja x_t so tako razporejena po časih, ko so bila zaznana, torej naraščajočem času t .

V realnosti pogosto velja, da so zaporedna opazovanja med seboj odvisna. To je predvsem očitno, če pomislimo na najbolj tipičen primer finančne časovne vrste, in sicer vrednostnega procesa cene neke delnice. Če vidimo, da ima v času t delnica ceno p , si lahko navadno mislimo, da bo v času $t + 1$ cena blizu p . Glede na čase, v katerih gledamo rezultate naše časovne vrste v grobem ločimo 3 tipe:

- Časovne vrste v zveznih časih. Tu pridobivamo podatke za vsak trenutek znotraj nekega časovnega intervala. Dober primer je tu EKG, kjer v vsakem trenutku preiskave naprava izrisuje graf električne napetosti proti času.

V analizi finančnih časovnih vrst za ta tip uporabljamo Black-Scholesov model. Gre za podvrsto parabolične diferencialne enačbe, katere rešitev nam da predvideno vrednost evropskih opcij *call* in *put* v času, ki nas zanima.

- Časovne vrste z diskretnimi vrednostmi kjer pridobivamo vrednosti, ki lahko zavzamejo le diskretne vrednosti, torej največ števno neskončno različnih možnosti.

Tipičen primer bi bil število klicev med 7.00 in 8.00 znotraj Ljubljane oziroma število nesreč znotraj enega dneva na avtocesti med Postojno in Ljubljano.

- Zadnja vrsta so diskretne časovne vrste, torej vrste, ki vrednosti vzamejo le v določenih časovnih trenutkih, npr. konec vsake ure, konec dneva, ... Tukaj lahko vrednosti zavzamejo kakršnokoli vrednost, pomembno je le, da te vrednosti vzamemo ob pravih trenutkih. Te vrste so uporabne tudi za aproksimiranje časovnih vrst v zveznih časih. Prav vrste takega tipa so tiste, ki jih lahko analiziramo z skritimi markovskimi modeli.

S finančnimi vrstami se srečujemo vsak dan in v najrazličnejših oblikah. Zaradi njihove raznolikosti je izbira pravega modela za njihovo analizo ključna. V našem primeru bomo za njihovo analizo uporabili skrite markovske modele.

Navadno je glavni namen analize časovnih vrst napovedovanje trendov, vendar se je

to področje z razvojem predvsem podatkovnega rudarjenja razširilo tudi na druga področja.

5.1. Dodatne lastnosti modela. Ko imamo opravka z modeli časovnih vrst, je zelo pomembno, da se zavedamo, da na vrednosti vpliva avtokorelacija. To pomeni, da nas zanima odnos, ki ga imajo zaporedna opazovanja. Da lahko o tem govorimo, pa moramo pogledati prvih nekaj momentov. Pri tem privzamemo sintakso iz 3.2.

Lema 5.3. *Naj bo f funkcija, za katero velja, da je $E(f(X)) < \infty$ za neko slučajno spremenljivko X . Naj bo Q_t slučajna spremenljivka, ki nam v času t pove opazovanje. Naj še velja, da je imamo glede na opazovanje možnih m opazovanj. Potem najprej velja, da je*

$$E(f(Q_t)) = \sum_{i=1}^m E(f(Q_t)|S_t = i)P(S_t = i)$$

poleg tega pa velja še

$$E(f(Q_t, Q_{t+k})) = \sum_{i,j=1}^m E(f(Q_t, Q_{t+k})|S_t = i, S_{t+k} = j)P(S_t = i)\Gamma_{ij}^k$$

kjer je Γ_{ij}^k ij element matrike Γ , kjer je Γ prehodna matrika med stanji.

Dokaz prvega je enostaven, saj $E(f(Q_t))$ zgolj pogojimo na $S_t = i$. Za drugo enakost pa moramo najprej pogojiti na S_{t+k} , potem pa upoštevamo, da je to pogojna verjetnost zgolj na $S_{t+k} = j$ in $S_t = i$. Na ta način smo prišli do izračuna drugega momenta, kjer za funkcijo f v prvem primeru vzamemo kvadriranje. Tako velja $E(Q_t^2) = \sum_{i=1}^m E(Q_t^2|S_t = i)P(S_t = i)$ in posledično je varianca modela enaka

$$Var(Q_t) = \sum_{i=1}^m E(Q_t^2|S_t = i)P(S_t = i) - \left(\sum_{i=1}^m E(Q_t|S_t = i)P(S_t = i)\right)^2.$$

Po drugi enačbi lahko izračunamo tudi $E(Q_t Q_{t+1})$, in sicer velja

$$E(Q_t Q_{t+k}) = \sum_{i,j=1}^m E(Q_t Q_{t+k})|S_t = i, S_{t+k} = j)P(S_t = i)\Gamma_{ij}^k.$$

Tako dobimo kovarianco kot

$$Cov(Q_t, Q_{t+k}) = E(Q_t Q_{t+k}) - E(Q_t)E(Q_{t+k}).$$

Ko imamo vse to podano lahko izračunamo Pearsonov korelacijsko funkcijo PCC za naše podatke. Velja da je PCC enak

$$\rho_k = \frac{Cov(Q_t, Q_{t+k})}{Var(Q_t)}.$$

Vrednost tega se tako nahaja med -1 in 1 , ki pomenita popolno koreliranost, oziroma popolno negativno koreliranost v primeru $\rho_k = -1$, medtem ko $\rho_k = 0$ pomeni, da podatki medsebojno niso korelirani.

5.2. Finančna časovna vrsta.

Definicija 5.4. Finančna časovna vrsta je časovna vrsta, kjer so opazovanja x_t vrednosti finančnega instrumenta v času t .

Finančna časovna vrsta je tako zaporedje opaženih vrednosti nekega finančnega instrumenta. Analiza finančnih časovnih vrst se ukvarja s teorijo in prakso določanja vrednosti finančnih instrumentov skozi čas. Zaradi elementa negotovosti, kot je na primer volatilitnost finančnih instrumentov, jo moramo obravnavati ločeno od vseh ostalih časovnih vrst. Velja namreč da ravno ta negotovost prikaže pomembnost statistične teorije in metod, izpeljanih iz le te, v analizi finančnih časovnih vrst. Z finančnimi časovnimi vrstami se ukvarjajo statistiki ter kvantitativni finančniki. V nasprotju z ostalimi finančnimi vrstami se lahko zaradi mikrostrukture trga zgodi, da imajo finančne časovne vrste posebne lastnosti in obliko. Tako se pogosto predpostavlja, da so donosi R normalno porazdeljeni z pričakovano vrednostjo μ in varianco σ^2 , $R \sim N(\mu, \sigma^2)$. Vendar to ne velja, saj lahko hitro ugotovimo, da donos ne bo nikoli manjši od -1 . Če predpostavimo, da je P_t cena instrumenta v času t , potem je očitno, da le ta nikoli ne bo manjša od 0. Potem je donos R_t definiran kot

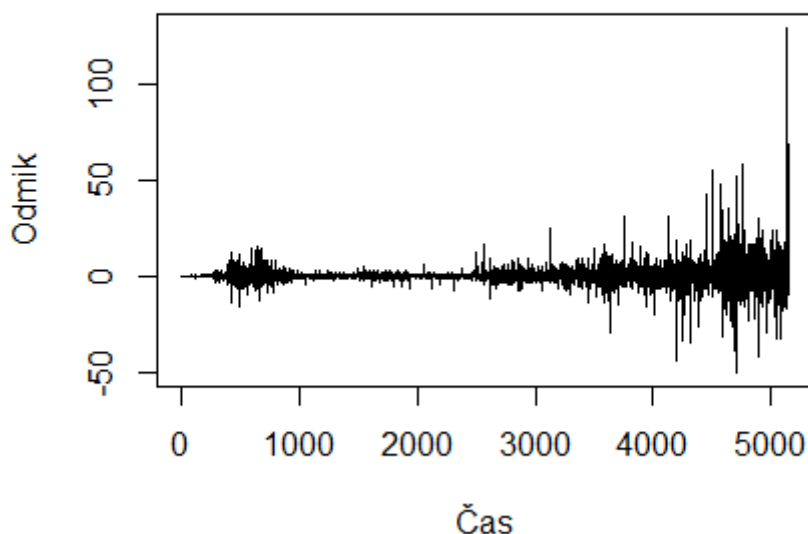
$$R_t = \frac{P_t - P_{t-1}}{P_{t-1}}$$

kar pomeni, da je

$$-1 = \frac{0 - P_{t-1}}{P_{t-1}} \leq R_t.$$

Ta problem v ekonomiji pogosto rešujemo z uporabo logaritmiranja. Alternativni pristop pa pravi, da vzamemo stacionarne vrste, kjer gledamo le relativne razlike med cenami delnic.

V prvi polovici 20. stoletja se je v povezavi z finančnimi trgi osnovala tako imenovana hipoteza o učinkovitem trgu, ki pravi, da če cene popolnoma predstavljajo pričakovanja in informacije vseh udeležencev trga, potem so njihove spremembe nepričakovane. To lahko povežemo tudi s teorijo slučajnih procesov. Teorija martingalov pravi namreč, da je najboljša ocena jutrišnje cene današnja cena. Zato v finančnih časovnih vrstah večjo težo damo podatkom, ki so nam zgodili kasneje.



SLIKA 4. Primer stacionarne vrste

6. UPORABA SKRITIH MARKOVSKIH MODELOV V FINANČNIH ČASOVNIH VRSTAH

Glavna ideja uporabe skritih markovskih modelov v finančni časovni sledi iz osnovne ideje Andrewa Viterbija 1, in sicer analiziranje signalov, ki jih prejmemo tako, da izločimo šume v prejetem signalu.

Aplikacija skritih markovskih modelov v finančne časovne vrste je sorazmerno enostavna. Naše izbrane podatke o ceni nekega instrumenta določimo kot zaporedje opazovanj O . Na podlagi tega določimo število stanj ter začetne ocene vseh parametrov.

V zadnjem času se je uporaba skritih markovskih modelov v finančništvu precej povečala zaradi široke uporabnosti, med drugim tudi za problem izbire portfelja. Gre za eno izmed klasičnih vprašanj, ki sem nam postavijo je v finančni optimizaciji. To vprašanje se da rešiti tudi z uporabo linearnega programiranja in skritih markovskih modelov. Uporabnost tega pristopa se pokaže predvsem zaradi možnosti omejevanja tveganja.

Primer 6.1. Problem izbire portfelja

Eno izmed klasičnih vprašanj, ki sem nam postavijo je v finančni optimizaciji je problem optimalne izbire portfelja. To vprašanje se da rešiti tudi z uporabo linearnega programiranja in skritih markovskih modelov.

Predpostavimo, da imamo v trenutnem času kapital M , ki ga lahko vložimo v N različnih vrednostnih papirjev. Odločiti se moramo, kako bomo naš kapital razdelili tako, da bomo maksimizirali svoj donos in hkrati minimizirali tveganje.

Vsak vrednostni papir ima donos R_j , kjer velja $j \in 1, \dots, N$. Za vsak j je donos slučajna spremenljivka, katere porazdelitev aproksimiramo z diskretno slučajno spremenljivko \hat{R}_j . Določimo še vektor $x = (x_1, \dots, x_N)$, kjer nam x_j pove, kakšen

delež kapitala smo vložili v posamezen vrednostni papir, to je

$$x_j = \frac{M_j}{M}$$

če velja $\sum_{j=1}^N M_j = M$. Ob koncu investicije bomo torej zaslužili

$$R_x = \sum_{j=1}^N x_j R_j$$

kjer je R_x slučajna spremenljivka.

To nalogo lahko predstavimo kot linearni program s pomočjo pogojne vrednosti ob padcu CVaR. CVar je mera tveganja, ki nam oceni tržno ali kreditno tveganje portfelja. Tako pridemo do optimizacijskega problema oblike

$$\begin{aligned} \min \quad & CVaR_\alpha(R_x) \\ \text{p.p.} \quad & E(R_x) \geq d \end{aligned}$$

kjer je d prej določen zahtevan donos, α pa stopnja zavrnitve.

Skriti markovski modeli se v igro vključijo, ko moramo aproksimirati donose. Kot že vemo, je \hat{R}_j diskretna aproksimacija donosa s končnim številom stanj. Ta stanja določimo kot različna možna stanja finančnega instrumenta na naših podatkih. Nato znanih podatkov določimo še vhodne in izhodne matrike za vse slučajne spremenljivke.

Z zbranim znanjem in podatki izvedemo simulacije več različnih scenarijev, čemur na koncu sledi še reševanje dejanskega linearnega programa. \diamond

Modeli, ki se jih uporablja v za napovedovanje cen finančnih instrumentov na kratki rok predvidevajo, da model sledi Brownovem gibanju, vendar velja, da ti modeli ne morejo zaznati ekstremnega gibanja cen. Tu je prednost tako imenovanih *regime-switching* modelov, torej modelov pri katerih se zamenjujejo stanja in med katere spadajo tudi skriti markovski modeli.

7. ZAKLJUČEK

V svoji seminarski nalogi sem opisal delovanje skritih markovskih modelov, pri čemer sem se osredotočil na finančne časovne vrste.

Skriti markovski modeli so spadajo med markovske modele, za katere velja, da velja markovska lastnost, katerih razvoj se je začel z Andrejem Markovom. Z njimi opišemo slučajne procese, pri katerih ne poznamo vseh parametrov, saj jih moramo iz podatkov še pridobiti.

Za vzpostavitev modela potrebujemo zaporedje opazovanj, iz katerih izvlečemo nabor parametrov λ , s katerimi poizkusimo maksimizirati funkcijo verjetja $P(O|\lambda)$. To funkcijo poizkusimo maksimizirati s pomočjo Baum-Welchovega algoritma, ki je ključni algoritem za delovanje skritih markovskih modelov.

Njihova uporaba je zelo široka, od procesiranje govora prek uporabe v biokemičnih procesih. Sam sem se posvetil uporabi v finančnih časovnih vrstah.

Časovne vrste so časovno urejena zaporedja opazovanj, v finančnih časovnih vrstah pa so ta opazovanja vrednosti finančnih instrumentov v času. Te vrste obdelujemo na več načinov, od ARIMA modelov do skritih markovskih modelov.

Finančne časovne vrste predstavljajo veliko različnih izzivov, med katerimi je med pomembnejšimi problem izbire portfelja, kjer si lahko pomagamo tudi s skritimi markovskimi modeli.

LITERATURA

- [1] D. Roman, G. Mitra in N. Spagnolo, *Hidden Markov models for financial optimization problems*, IMA Journal of Management Mathematics **21** (2010) 111–129.
- [2] I.L. MacDonald in W. Zucchini, *Hidden Markov and Other Models for Discrete-valued Time Series*, Chapman & Hall/CRC Monographs on Statistics & Applied Probability **70**, Chapman & Hall, London, 1997.
- [3] R.S. Mamon in R.J. Elliott *Hidden Markov Models in Finance*, International Series in Operations Research & Management Science **104**, Springer, New York, 2007.
- [4] P.J. Brockwell, R.A. Davis *Introduction to Time Series and Forecasting*, 2nd edition, Springer, 2002.
- [5] B.J. Yoon, *Hidden Markov Models and their Applications in Biological Sequence Analysis*, v: Current genomics, 10,6(2009):402-415, [29. 7. 2019], dostopno na <https://www.ncbi.nlm.nih.gov/pmc/articles/PMC2766791/>.
- [6] *A Hidden Markov Model method, capable of predicting and discriminating β -barrel outer membrane proteins*, BMC Bioinformatics **5** (2004)
- [7] P.G. Bagos, T.D. Liakopoulos, I.C. Spyropoulos, S.J. Hamodrakas, *A Hidden Markov Model method, capable of predicting and discriminating β -barrel outer membrane proteins*, v: BMC Bioinformatics, 5(2004), [30. 7. 2019], dostopno na <https://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-5-29>.
- [8] P. Dymarski, *Hidden Markov Models, Theory and Application*, InTech, Rijeka, 2011
- [9] A. Geyer, W. T. Ziemba, (2008) *The Innovest Austrian Pension Fund Financial Planning Model InnoALM*. Operations Research 56(4):797-810.
- [10] C.E. Shannon, *A Mathematical Theory of Communication*. Bell System Technical Journal. 27 (3): 379–423
- [11] A.A. Markov, (1906) *Extension of the Law of Large Numbers to Dependent Events*. Bulletin of the Society of the Physics Mathematics, Kazan, Russia, 2, 155-156.
- [12] A.A. Markov, *Primer statisticheskogo issledovaniya nad tekstom "Evgeniya Onegina", illyustriruyushchij svyaz' ispytaniy v cep'*, Izvestiya Akademii Nauk, SPb, VI seriya, tom 7, 9 3, 1913, 153-162.
- [13] A. Dempster, N. Laird, D. Rubin, (1977). *Maximum Likelihood from Incomplete Data via the EM Algorithm*. Journal of the Royal Statistical Society. Series B (Methodological), 39(1), 1-38.
- [14] A. Viterbi, (1967). *Error bounds for convolutional codes and an asymptotically optimum decoding algorithm*, IEEE Transactions on Information Theory. 13, Nr. 2, pp.260-269.
- [15] L.E. Baum et al. (1970). *A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains*, The Annals of Mathematical Statistics, Vol. 41, No. 1, 164–171.
- [16] R. L. Stratonovich, *Conditional Markov Processes*, Theory Probab. Appl., 5:2 (1960), 156–178.
- [17] J.D. Hamilton, *A new approach to the economic analysis of nonstationary time series and the business cycle*, Econometrica: Journal of the Econometric Society (1989), 357-384.
- [18] D. Holmes, (2004, 9. 11). *What is Stochastic Programming*, dostopno na <http://users.iems.northwestern.edu/~jrbirge/html/dholmes/StoProIntro.html>.
- [19] J. Joyce, (2003,28.6). *Bayes Theorem*, dostopno na <https://stanford.library.sydney.edu.au/entries/bayes-theorem/>.